ABSTRACT

Title of dissertation: A CONTINUOUS-TIME, NONLINEAR OBSERVER FOR ESTIMATING STRUCTURE FROM MOTION FROM OMNIDIRECTIONAL OPTIC FLOW

Joseph K. Conroy, Doctor of Philosophy, 2010

Dissertation directed by: Professor J. Sean Humbert
Department of Aerospace Engineering

Various insect species utilize certain types of self-motion to perceive structure in their local environment, a process known as *active vision*. This dissertation presents the development of a continuous-time formulated observer for estimating structure from motion that emulates the biological phenomenon of active vision. In an attempt to emulate the wide-field of view of compound eyes and neurophysiology of insects, the observer utilizes an omni-directional optic flow field. Exponential stability of the observer is assured provided the persistency of excitation condition is met. Persistency of excitation is assured by altering the direction of motion sufficiently quickly. An equal convergence rate on the entire viewable area can be achieved by executing certain prototypical maneuvers. Practical implementation of the observer is accomplished both in simulation and via an actual flying quadrotor testbed vehicle. Furthermore, this dissertation presents the vehicular implementation of a complimentary navigation methodology known as *wide-field integration* of the optic flow field.

The implementation of the developed insect-inspired navigation methodologies on physical testbed vehicles utilized in this research required the development of many subsystems that comprise a control and navigation suite, including avionics development and state sensing, model development via system identification, feedback controller design, and state estimation strategies. These requisite subsystems and their development are discussed.

A CONTINUOUS-TIME, NONLINEAR OBSERVER FOR ESTIMATING STRUCTURE FROM MOTION FROM OMNIDIRECTIONAL OPTIC FLOW

by

Joseph K. Conroy

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2010

Advisory Committee:
Professor Dr. J. Sean Humbert, Chair/Advisor
Professor Dr. Darryll Pines, Co-Advisor
Professor Dr. Inderjit Chopra
Professor Dr. Robert Sanner
Professor Dr. Rama Chellappa, Dean's Representative

Acknowledgments

Firstly, I would like to thank both of my advisors, Dr. J. Sean Humbert and Dr. Darryll Pines. Dr. Darryll Pines took me on originally as an undergraduate student and then provided me an opportunity to attend graduate school in spite of a questionable academic record. I hope that what I tried to contribute back to the school was sufficient to justify his original leap of faith in me. Once hired to further micro air vehicle autonomy, Dr. Pines allowed me to work with minimal interruptions as I spent several years, necessarily, to teach myself electronics development, microcontroller programming and other hardware-oriented prerequisites that did not exist in our department at the time. Dr. J. Sean Humbert became my advisor later on in the program and provided me with the means to complete my dissertation research, both financially through the ever-necessary equipment purchases and mentally through his infectious enthusiasm for success, accomplishment, and winning in general.

I would also like to thank the rest of my committee, Dr. Robert Sanner, Dr. Rama Chellappa, and Dr. Inderjit Chopra for their time and advice. Each was happy to spend time to talk with me about my research in spite of their busy schedules.

I would like to acknowledge Geof Barrows, Craig Neely, and Travis Young at Centeye, Inc. who were a pleasure to work with from the beginning.

So much of the research that is accomplished in the AVL is a team affair. It has been an enjoyable experience to work (and play) with everyone in the department.

The fantastic level of cooperation within the lab has made everyone work more efficiently and made the social experience (such as it is being a grad student) way better. I hope that the cohesion that currently exists is not lost.

It took me 7 years to complete this degree, which is 2 more than you are supposed to take, (at least according to some stupidly optimistic engineering school goal). Nevertheless, my graduate school social experience will forever be defined by these last two years. Greg, I enjoyed our coming-of-age social experiences going to Cornerstone and the Turtle. Badri, you are one of the hardest working guys I've ever known. I'll one day miss our occasional walks around McKeldin Mall. Andrew "Slip-Slop" Hyslop, you're passion for experiencing everything new is refreshing. I will try to have that outlook also now that I will have more free time on my hands. Renee, always stay as happy-go-lucky and optimistic as you are. Also big shout-outs to Mac, Derek, Steve, Imraan, Kedar, the two new guys Nick K. and Doug, and the other AVL folks for all bringing life to the lab.

A HUGE "Thank You!" to my parents for their support not only through this endeavor, but throughout my life. They understood the sacrifice required of time and attention to complete this degree and were behind me the whole time.

I would like to thank the Lord Jesus Christ for providing me with the opportunity to complete this, and I hope I can use this knowledge to serve him and others after such a long, yet necessary, period of selfish isolation for study.

Last and also least, I would like to acknowledge coffee, which I have developed an unwelcome addiction to, and the energy drinks that got me through the overnighters. I'm switching to tea.

Table of Contents

List of Tables

List of Figures

Nomenclature

x	Inertial x location
y	Inertial y location
z	Inertial z location
u	Body-fixed longitudinal velocity
v	Body-fixed lateral velocity
w	Body-fixed heave velocity
p	Body-fixed roll rate
q	Body-fixed pitch rate
r	Body-fixed yaw rate
ϕ	Roll angle
θ	Pitch angle
ψ	Yaw angle
Ω_{mr}	Main rotor rotational speed
Ω_{tail}	Tail rotor rotational speed
μ_{lat}	Vehicle lateral input
μ_{lon}	Vehicle longitudinal input
μ_y	Vehicle yaw input
μ_t	Vehicle throttle input
R_{mr}	Main rotor radius
R_{tail}	Tail rotor radius
c_{mr}	Main rotor blade chord
c_{tail}	Tail rotor blade chord
$C_{d_{mr}}$	Main rotor drag coefficient
$C_{l\alpha}$	Tail rotor lift coefficient
ρ	Air density (Chapter 4 only)
l_{tail}	Tail moment arm
N_r	Yaw damping derivative
N_{μ_y}	Yaw control sensitivity
a	Longitudinal rotor flapping state (Chapter 4 only)
b	Lateral rotor flapping state
$\Delta t_{\mu_{cyclic}}$	Pure delay in seconds for both cyclic actuators
Δt_{μ_t}	Pure delay in seconds for throttle actuator
Δt_{μ_y}	Pure delay in seconds for yaw actuator
τ_f	Rotor time constant
B_a	Longitudinal flap angle to lateral flap rate derivative
L_b	Lateral flap angle to roll rate derivative
L_v	Lateral velocity to roll rate derivative
A_b	Lateral flap angle to longitudinal flap angle derivative
M_a	Longitudinal flap angle to pitch rate derivative
M_u	Longitudinal velocity to pitch rate derivative

N_r	Yaw rate derivative
Z_w	Heave velocity derivative
$Z_{\Omega_{mr}}$	Main rotor speed to heave velocity derivative
Y_v	Lateral aerodynamic damping derivative
X_u	Longitudinal aerodynamic damping derivative
$T_{\Omega_{mr}}$	Main rotor speed damping derivative
L_ϕ	Roll angle to roll rate damping derivative
M_θ	Pitch angle to pitch rate damping derivative
L_{lat}	Lat. input to roll rate sensitivity
M_{lon}	Lon. input to pitch rate sensitivity
g	Gravity constant
B_{lat}	Lateral input to roll rate sensitivity
B_{lon}	Longitudinal input to pitch rate sensitivity
A_{lat}	Lateral input to pitch rate sensitivity
A_{lon}	Longitudinal input to roll rate sensitivity
N_{μ_y}	Yaw input to yaw rate sensitivity
T_{μ_t}	Throttle input to main rotor speed sensitivity
$\mathbf{x}$	Pose state vector
z	Linearized output (Chapter 6 only)
a	Tunnel half-width (Chapter 6 only)
$N_{\mu_{yaw}}$	Effective yaw damping (honeybee vehicle)
$\mu(\gamma)$	Nearness function
γ	Retinal viewing angle
ψ	Inertial vehicle orientation
ω	Vehicle yaw rotation rate
$d(\gamma)$	Distance function
$\lambda(\gamma)$	Velocity distribution
ρ	Observer gain (Chapter 7 only)
α_k	Spatially discretized observer convergence rate

MAV	Micro air vehicle
VLSI	Very large scale integration
OF	Optic flow
DSP	Digital signal processor
STFT	Short time Fourier Transform
GPS	Global Positioning System

Chapter 1

Introduction

Over the past several decades, Unmanned Aerial Vehicles (UAVs) have seen increasing use in military [2, 3] operations and have the potential to greatly affect civilian applications [4]. More recently, a new interest has arisen regarding the development of a smaller class of flying robotic platforms, known as micro air vehicles (MAVs). Whereas larger UAVs are designed for missions at high altitude, MAVs are designed to operate lower to the ground, capable of navigating complex outdoor and indoor environments filled with obstacles with stealth, low cost, and low operator workload. Ideally, it is envisioned that these vehicles will operate in swarm formations as a single organic entity capable of performing missions such as surveillance, chemical plume detection, and radiation detection which no single vehicle would be capable of alone.

The ability of a MAV to navigate through an unknown environment requires that it be capable of estimating distance to relevant obstacles in the environment, or perhaps at minimum, detecting unsuitable proximity to these obstacles. The ability to sense and detect obstacles surrounding the vehicles reliably is currently considered an advanced capability, even for larger vehicles.

Vision is an attractive method for providing environmental awareness, as the visual field is a rich source of information about the surrounding environment. Tradi-

tional machine vision approaches [5, 6, 7, 8, 9, 10] that infer proximity and velocity information from camera imagery have been demonstrated. These methods have been computationally expensive and typically require off-board visual processing, even on vehicles with significant payloads [11]. Furthermore, these implementations of navigation tend to result in behavior where their speed and direction is limited not by the dynamics of the vehicle, but by the obstacle avoidance processing speed and field of view. The vehicles that currently utilize vision tend to be large as they must accommodate the necessary hardware and sensing for machine vision, and must move slowly enough for the algorithms to be effective.

Operating in this manner is clearly an impediment to the spirit of the operation of micro air vehicles. Drifting unnecessarily slowly towards a target objective would be a waste of propulsion energy, as would lifting the bulky processing and sensing hardware, and would not make use of the maximum flight envelope the vehicle is capable of. Furthermore, as the vehicle size decreases, maneuverability and overall quickness tends to increase, thus the capability gap between the responsiveness of a given MAV and current typical navigation solutions becomes even more apparent.

The limited payload of MAVs, with the additional requirement that the visual system be capable of reacting quick enough to allow the vehicle to fly within its full flight capabilities remains a challenge. New ways of approaching the visual navigation problem are required to provide the necessary capabilities within a reasonable hardware and energy footprint.

Fortunately, nature provides an excellent example of robust visual perception, as flying insects are capable of navigation within uncertain environments without the

computational complexity that current machine vision algorithms require to perform the same tasks. Recent studies at the behavioral and neurophysiological levels have provided insight on the neural mechanisms that give rise to these sophisticated capabilities. Insects encode optic flow [12], the characteristic patterns of visual motion that form on their retinas as they move.

Specialized neurons called tangential cells [13], parse these complex optical flow patterns over large swaths of the visual field to extract visual cues for navigation, such as visual odometry. Furthermore, biological organisms are capable of detecting and tracking small targets, particularly conspecifics, comparatively swiftly and easily even against highly textured backgrounds[14, 15, 16].

Finally, they are able to perform prey pursuit interception[17] and predator evasion through motion camouflage[18], and structure identification from motion[19]. Compared to current robotics technology, these are amazing feats and we wish to replicate, or at least emulate to provide the next generation of micro air vehicles with the most advanced navigation systems possible.

1.1 MAV System Design

The development of a MAV guidance, navigation, and control system requires forethought and planning to prevent unnecessary bottlenecks in performance. There are many components of such a system that need to work together at once and must be matched to do so efficiently. Figure 1.1 shows the typical architecture of a control and navigation system utilizing feedback. It is desired that the vehicle's *actual*

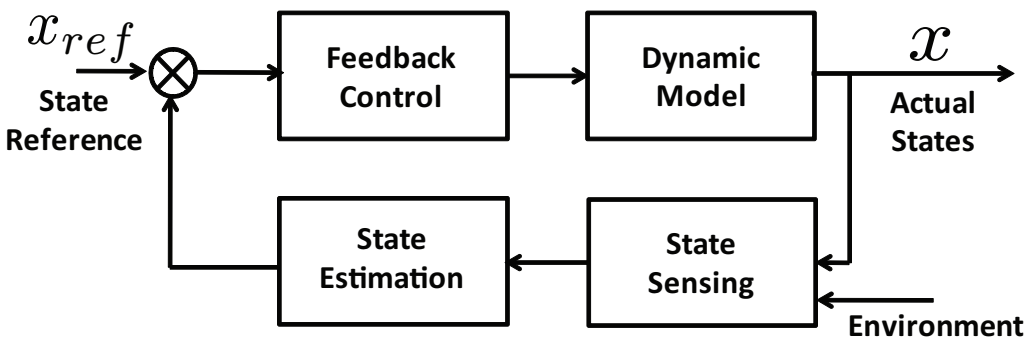

Figure 1.1: Typical Navigation and Control Architecture using Feedback

motion, x, closely track some desired set of states or criteria x_{ref}. The feedback control law specifies how to change the vehicle's control inputs to best accomplish this tracking. Appropriate design of the feedback control law assumes reasonable knowledge of how the vehicle moves in response to these inputs, given by the dynamic model, typically represented a set of differential equations. The feedback control law requires knowledge of certain states to ensure good tracking. If a particular required state is not directly measured by a sensor, it must be estimated using the sensors that are available, suggesting the need for a state estimation methodology as well.

The performance of the entire system relies on adequate performance of each of the components in figure 1.1, therefore MAV control and navigation design should be approached from a systems-level design perspective. Improvements to overall system performance can result from more efficient processing of sensory information, better control via a better understanding of vehicle dynamics, and designing the overall control and navigation architecture such that the stability and navigation control methods are closely intertwined.

To address these points, this research is dedicated to advancing the state of the art in biologically-inspired visual navigation while also specifically addressing the requisite design aspects of avionics design, sensor integration, state estimation, feedback control and utilizing system identification as a practical method of MAV dynamic model establishment, as the performance of the visual control is dependent on these factors performing adequately as well.

1.2 Vehicle Testbeds

To demonstrate the practical and realistic feasibility of the developed theory, research in this thesis was physically implemented on flying vehicles when possible. Two separate vehicle testbeds were utilized for this purpose. The first vehicle is a single main rotor electric hobby helicopter(Esky), figure 1.2, that was retrofit to include the sensors and avionics required. The second was a commercially available quadrotor vehicle(Ascending Technologies) that was also retrofit.

1.3 Thesis Contributions and Organization

This dissertation is organized as follows. Chapter 2 presents the development of a prototype custom avionics and the integration of optic flow sensors designed specifically to support laboratory testing of the developed navigation algorithms. Chapter 3 presents useful on-board state estimation methods that have been implemented, with a focus on estimating attitude dynamics. Chapter 4 presents how system identification techniques can be used to efficiently develop reasonable ve-

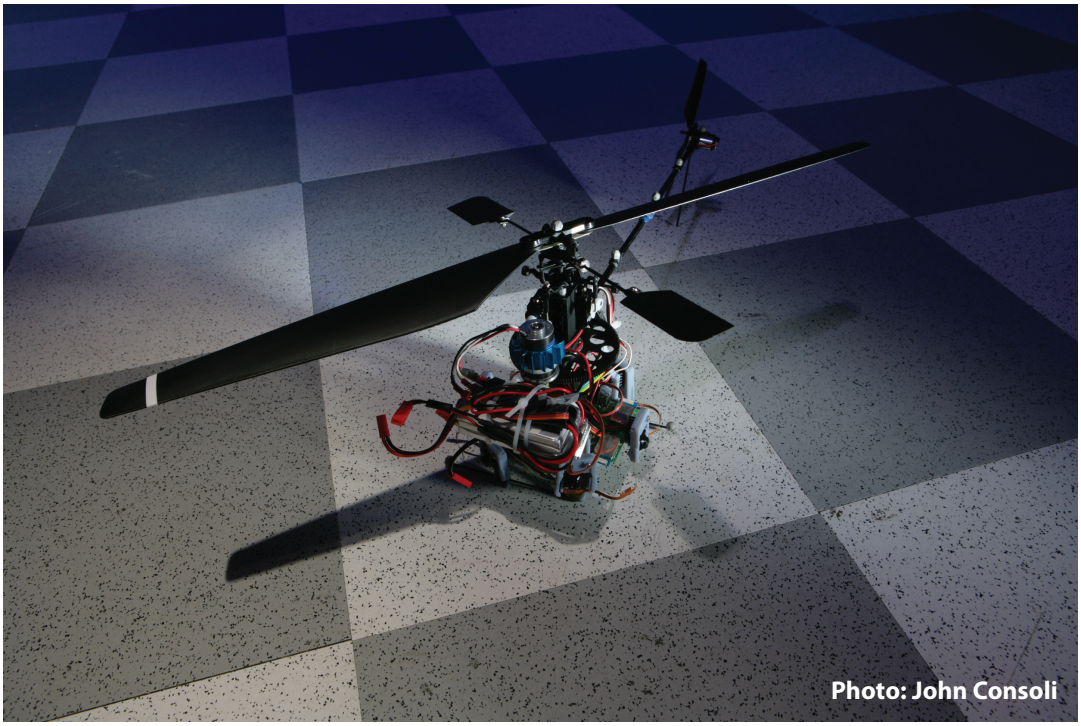

Figure 1.2: Honeybee Testbed Vehicle

Figure 1.3: Quadrotor Testbed Vehicle

hicle models and attain insight into vehicle behavior. The system identification and subsequent analysis of vehicle dynamics of a single main rotor micro helicopter is presented. Chapter 5 discusses the implementation of both velocity regulated and position regulated feedback control, typically used to augment the visual stability algorithms. This chapter focuses on feedback that is directly unrelated to, yet supports, the subsequent visual navigation algorithms. Chapter 6 presents the implementation of the *wide-field integration*(WFI) navigation methodology, a prior developed biologically inspired technique enabling the extraction of pose information from optic flow. Chapter 7 presents a continuous-time nonlinear observer for structure from motion utilizing omni-directional optic flow. A Lyapunov analysis is used to prove stability. The persistency of excitation requirement for the observer is compared to the prototypical motions exhibited by insects for the purpose of environmental structural identification through *active vision*. Chapter 8 discusses the conclusions of this research and presents some directions for future work.

Chapter 2

Micro Air Vehicle Sensing and Avionics

This chapter presents the development of micro air vehicle avionics that are custom designed for research and development. For any actual vehicle implementation, the avionics sensing and processing capabilities limits the possible theory and algorithms that can be practically implemented. Vehicles of the target MAV weight have limited payload capability and sensor accuracy is usually the tradeoff to obtain lighter components. Nevertheless, care must be taken in the design to minimize the reduction in theoretically attainable system performance caused by avionics design tradeoffs.

2.0.1 Commercially Available Autopilots

A number of autopilots have been developed, both by research institutions and commercially, for enabling fixed-wing and rotary-wing autonomous flight. End user solutions typically utilize built-in estimation algorithms to provide state estimates. The usage of inertial sensing for attitude estimation is fairly ubiquitous. Most current designs also rely on accommodating GPS and pressure sensing to provide translational motion estimation. A comparison of currently available commercially available avionics is shown in table 2.1.

Current methods of sensing translational motion are not suitable for slow-

Company	Model	Weight	Sensors	Wireless Telemetry
Procerus	Kestrel v2.4	16.7 gm.	triaxial accel, rate, mag	none built-in
Micropilot	MP2128	28 gm.	triaxial accel, rate	100 Hz max
O_NAVI	PhoenixAX	43.5 gm.	triaxial accel, rate	100 Hz (separate kit)
Continental Control and Design	Tiny Guidance Engine	13 gm.	triaxial accel, rate, mag	50 Hz (separate kit)
UMD AVL Avionics	v3.4	16.9 gm.	triaxial accel, rate, mag	>300 Hz accel/rate, ∼30 Hz Mag

Table 2.1: Avionics Comparison

speed flight in highly cluttered environments. GPS cannot be relied upon indoors, and pressure sensing does not provide velocity or height estimates at a sufficient resolution for the low range of motion of the test vehicle. Furthermore, some method of being able to detect proximity to surrounding obstacles is required. This reinforces the primary motivation for the development of vision-based translational motion sensing and obstacle avoidance. Traditional machine vision is difficult to implement as the necessary processing hardware is prohibitively heavy and power hungry. To fully utilize the benefits of vision while permitting operation on small palm-sized vehicles, new ways of approaching the sensing and processing are required.

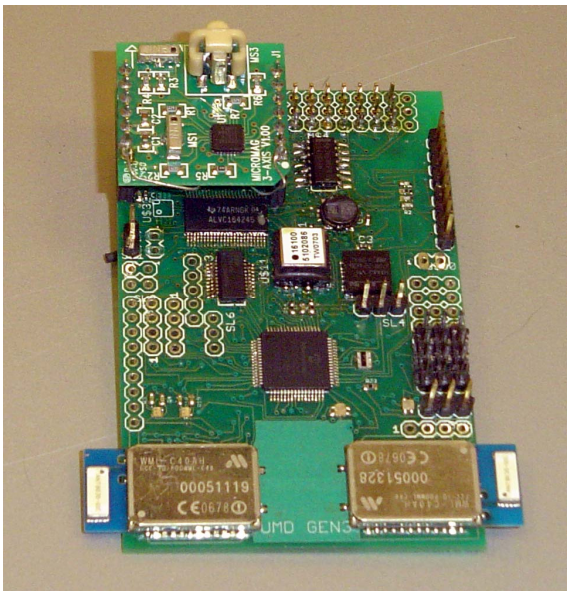

Figure 2.1: UMD Generation 3.1 Avionics Board

2.0.2 MAV Avionics Board

To facilitate the research described in the following chapters, a highly reconfigurable avionics package was developed and fabricated[20]. A primary motivator for a custom design has been to accommodate VLSI-based optic flow sensors designed by Centeye, Inc. The board, shown in figure 2.1, weighs under 15 grams, and includes triads of accelerometers, rate gyros, and magnetometers. Two 8-bit PIC microcontrollers enable on-board processing tasks, one for sensor data acquisition and communications interfacing, and one for the time critical on-board feedback control calculations and actuator outputs. The board is capable of interfacing with up to 8 Radio Control (RC) servos or RC motor speed controllers. Two Class 1 Bluetooth radio modems(Blueradios) are integrated that can be used for uplinked commands and telemetry data. The Bluetooth module has a limited communications range of less than 100 meters line-of-sight, however it was selected for the

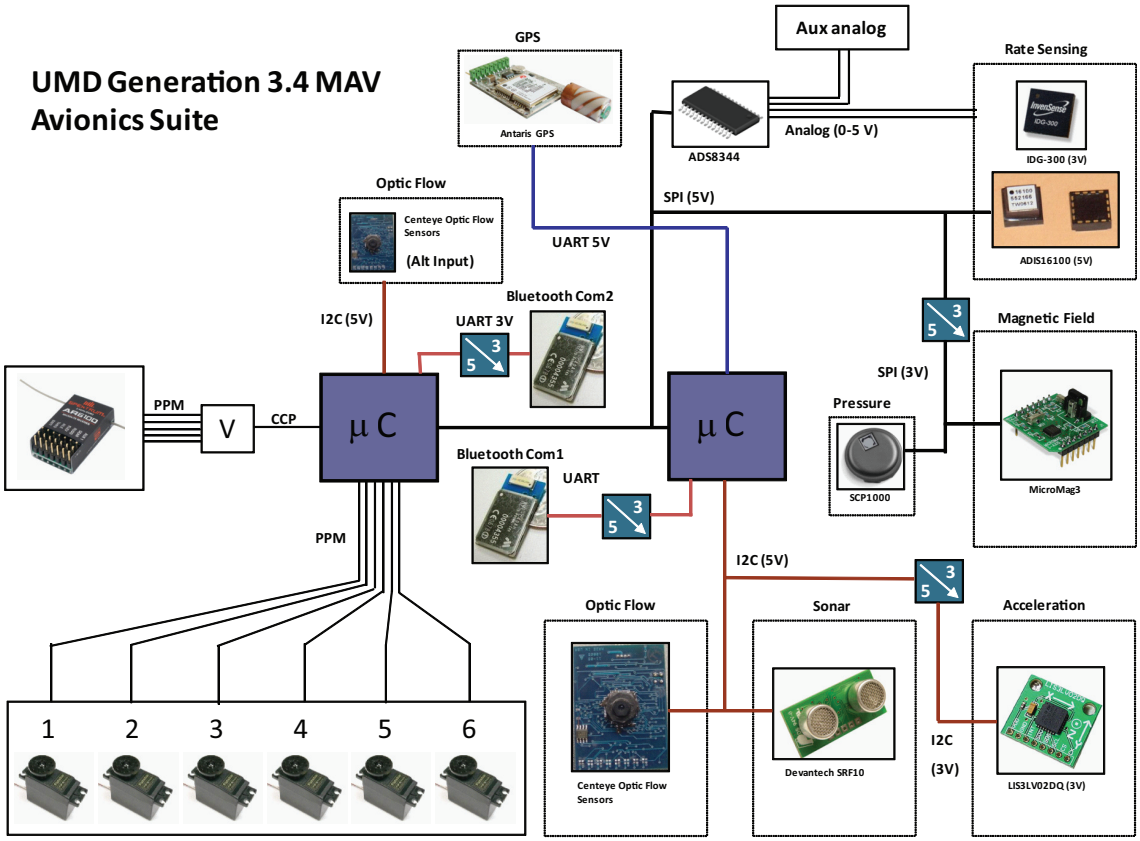

Figure 2.2: UMD Generation 3.4 Avionics Board Diagram

capability of providing high bandwidth telemetry. Although each modem is capable of bi-directional communication, two modules were used in single direction communication to minimize the random byte gap delays inherent to full duplex (bi-directional) communications over a single module. The Bluetooth connection is capable of real-time telemetry of on-board data at a rate that is currently unmatched among commercially available competitors in the same weight class. Additionally, the board has an input port that allows a user to send commands using a traditional RC receiver. The flexibility of the I^2C bus, originally utilized for optic flow sensor integration, has also been used for interfacing with a sonar range finder, useful for altitude measurements.

2.0.3 Calibration

Calibration of the rate gyros was performed using a rate table. The accelerometers were calibrated by successively rotating the unit at orthogonal angles to the known gravity vector. The magnetometer triad required additional consideration for calibration. The direction of the magnetic field varies at different locations on Earth, and the magnetic field is generally not orthogonal to the gravity vector. Furthermore, the local magnetic field is easily affected by *hard iron* errors and *soft iron* errors. Hard iron errors are local materials in the building on on the vehicle itself that introduce a constant additive local field. Soft iron errors do not create a local magnetic field, but rather distort the field such that the error might be orientation dependent.

To minimize the effects of these errors, it is best if calibration is performed in the area of operation with the sensors mounted on the vehicle in the final configuration in which it is to be operated. This type of in-situ calibration can be performed using nonlinear estimation methods. One on-line estimation procedure is presented in [21, 22] and experimentally verified in [23]. As the sensor unit is rotated, it is expected that the magnitude of the sensor triad stay constant, essentially sweeping out a sphere in 3D space. This same calibration procedure can be utilized for accelerometers as well as the magnetometers without modification. The magnitude of the sphere would be the magnitude of gravity in the accelerometer case and the magnitude of the local magnetic field in the magnetometer case.

2.1 Optic Flow Sensing

Optic flow is defined as the apparent motion of the visual field perceived by an observer in response to its rotation and translational motion through the imaged environment[24, 25]. In the general sense, the optic flow field in an image of the environment is a 2D distribution of apparent velocities that can be associated with time varying changes in the intensity field of the image[26]. Optic flow does not require following specific feature points within the visual field, but rather can be simply thought of as the ratio of the temporal change in image luminance to the spatial luminance rate of change. Optic flow is quantified in units of $radians/sec$ as the optic flow generated by an object perceived by an observer appears as an angular change in the location of that object. Optic flow is the primary method of visual perception that many insects employ. Insects compute optic flow continuously over their entire field of view using spatio-temporal correlation neurons[27].

2.1.1 VLSI Optic Flow Sensing

Optic flow can require significant processing capabilities to compute quickly given raw images, however if the field is preprocessed using VLSI circuitry, much of computation can be performed in the analog domain and in parallel. Optic flow can then be estimated from the spatio-temporally preprocessed visual array quickly using a digital signal processor (DSP) or equivalent. The integration of a single optic flow sensor and sonar distance sensor is shown in figure 2.3.

The I^2C bus on the avionics board supports dozens of sensors. For optic flow

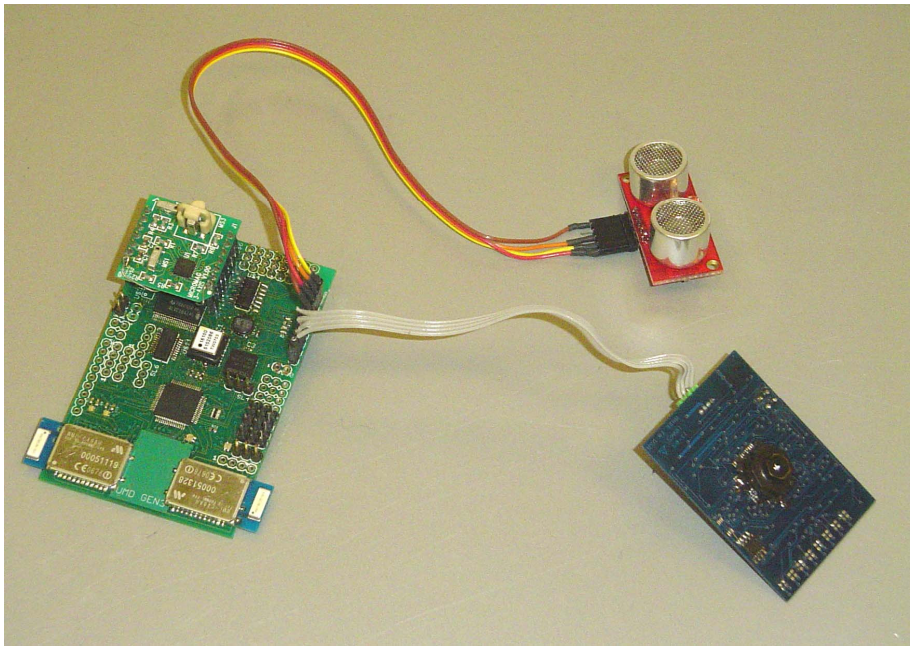

Figure 2.3: Single Optic Flow Sensor System Setup.

coverage around the azimuth of the Honeybee vehicle, a sensor ring was developed, shown in 2.4, comprised of 6 individual sensors. This concept was later upgraded to a sensor ring comprised of 8 Centeye Faraya optic flow sensors, figure 2.5, for use with the Quadrotor vehicle.

2.1.2 Omni-directional Vision Pod

As an alternative to the VLSI sensors, an omni-directional vision pod was developed, based on a commercially available camera board (Surveyor), consisting of a 500 MHz Blackfin fixed point processor (Analog Devices), which interfaces to a camera sensor (Omnivision OV 7725). The camera captures images of size 160 $\times$ 120 at 55 fps. A 360 degree field of view is obtained from a parabolic mirror installed above the quadrotor, with the camera pointing upward facing the mirror

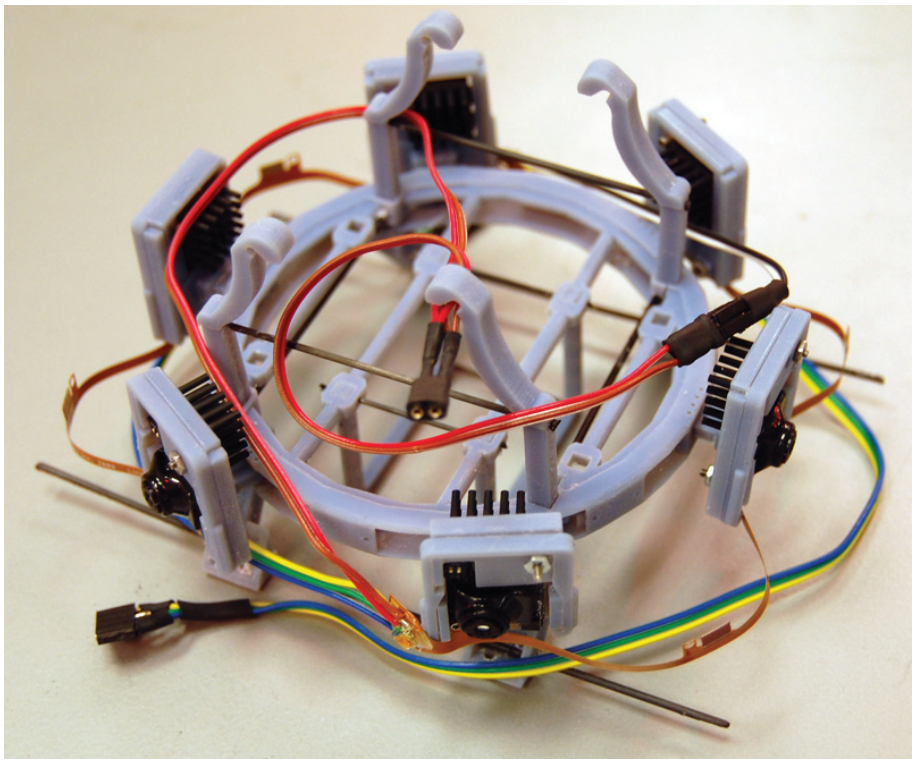

Figure 2.4: Sensor ring composed of 6 Centeye "ARZ-lite" sensors.

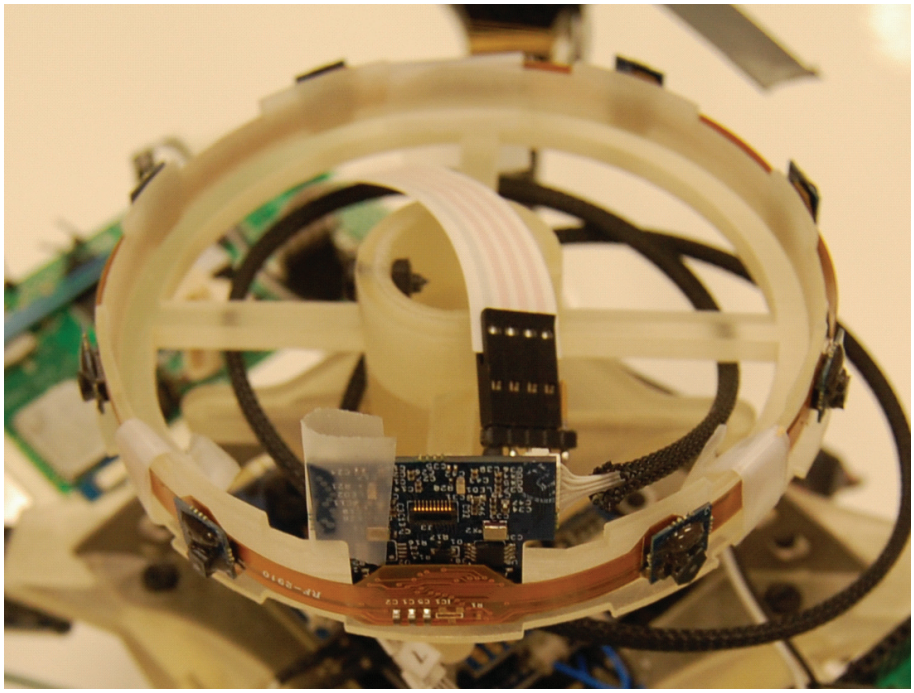

Figure 2.5: Sensor ring composed of 6 Centeye "Faraya" sensors.

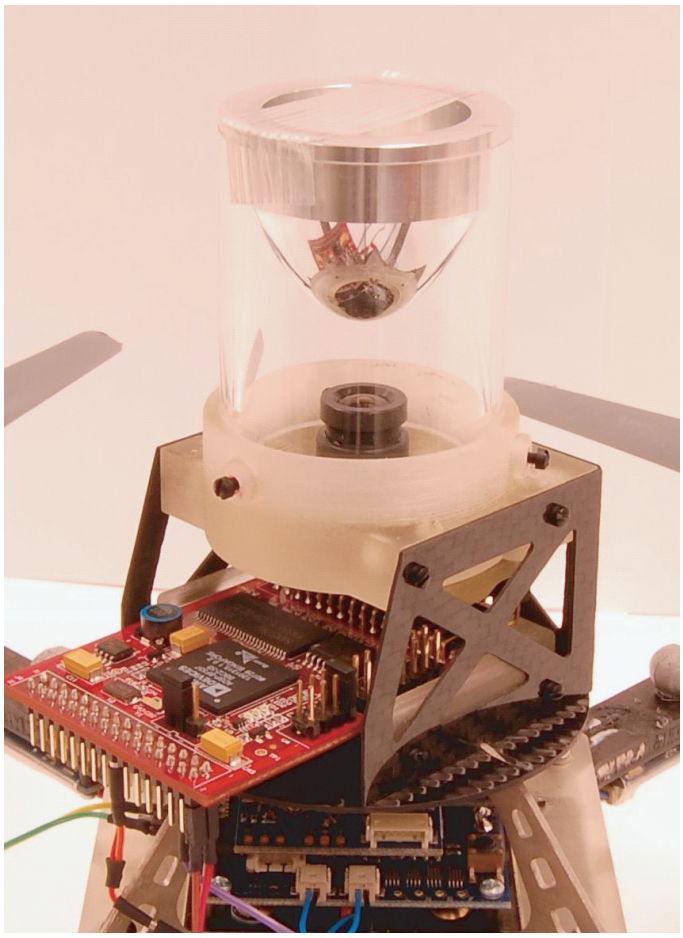

Figure 2.6: Omni-directional image processing pod.

(Figure 2.7).

The firmware contains code for performing optic flow computations and wide-field integration. The optic flow is computed using a gradient based method [28]. Since the processor is fixed-point, all computations were performed with scaled integers.

The computed optic flow from two successive grayscale image frames determines the motion field around the azimuth of the vehicle. To compute this azimuthal optic flow efficiently, the algorithm tracks only the movement of a set of 1000 tar-

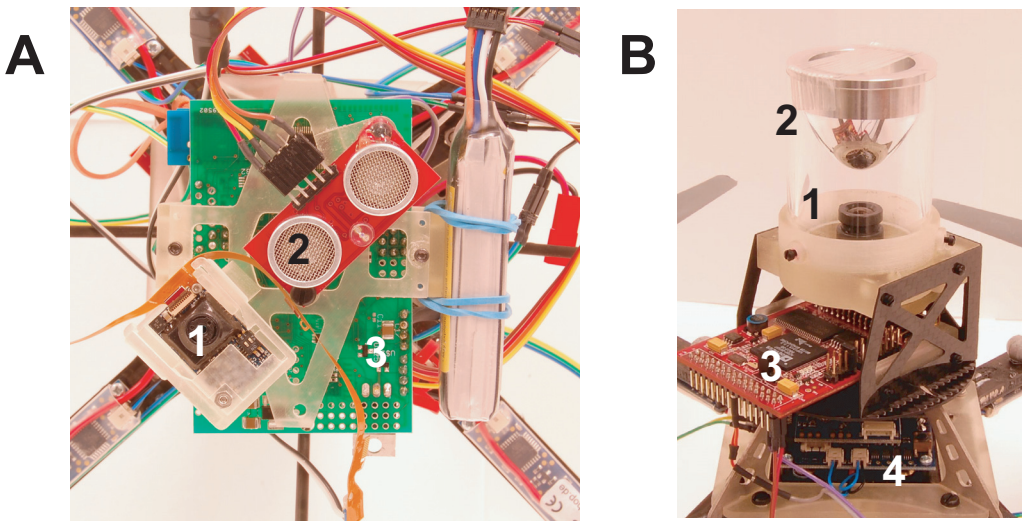

Figure 2.7: (A) Bottom View: (1) Centeye Optic Flow Sensor, (2) Sonar Sensor, (3) In-House Avionics, (B) Side View: (1) Surveyor Camera, (2) Parabolic Mirror, (3) Blackfin Processor, (4) X-UFO Proprietary Avionics.

get pixels, all located in one of four concentric rings of pixels at fixed radii from the mirror center (Figure 2.8). Each ring captures a different line of approximately constant height. Measurements from four rings are averaged vertically to improve performance for wall regions with poor contrast. The component of the shift tangent to the ring is used in the controller. By taking the dot product of the shift vector with the ring tangent vector, an estimate is obtained of the 1-D optic flow at 20-220 discrete angular azimuthal locations.

2.2 Off-Board Kinematic Motion Capturing

To provide an accurate measurement of the vehicle kinematics, a Vicon visual tracking system, seen in Figure 2.9, was used to obtain direct estimates of the position and orientation of the vehicle at 350 Hz. The system operates by using a

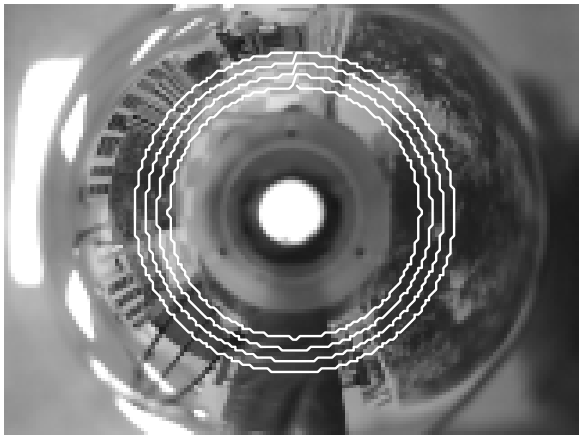

Figure 2.8: Optic flow sampling rings in parabolic mirror.

set of 8 high speed cameras to triangulate and track retro-reflective markers affixed to the vehicle. A collection of markers fixed relative to each other constitutes a rigid body and allows the determination of vehicle orientation.

This direct measurement of vehicle inertial-referenced position and orientation was valuable for system identification, feedback control and trajectory following verification, among other things. For system identification, a direct measurement of the inertial position and orientation improves the overall quality of the identification, as this avoids the use of state estimation algorithms that may introduce artificial observer dynamics that would need to be considered. Likewise, for feedback control, availability of a direct Euler angle estimate allows a comparison with an on-board IMU estimated orientation. Similarly, an inertial position estimate allows regulation of a vehicle to a specific point in the test area, a feature that is difficult to replicate without advanced visual processing.

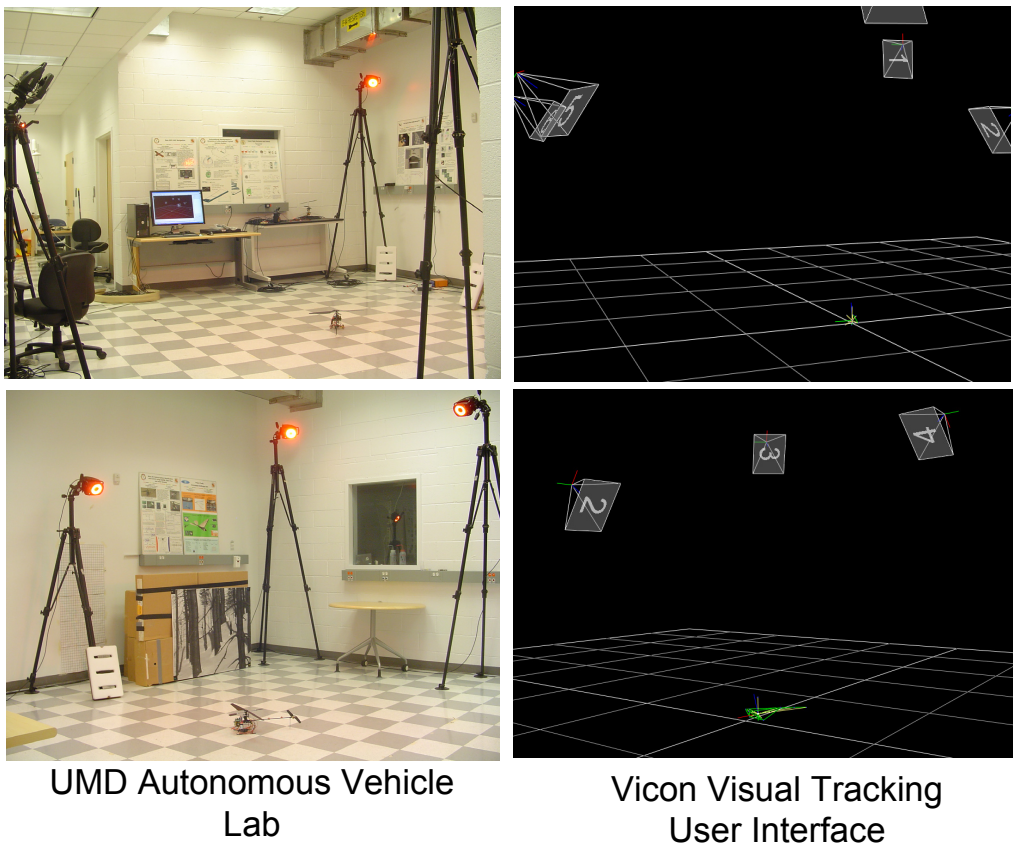

Figure 2.9: UMD Autonomous Vehicle Lab and Vicon Tracking Interface

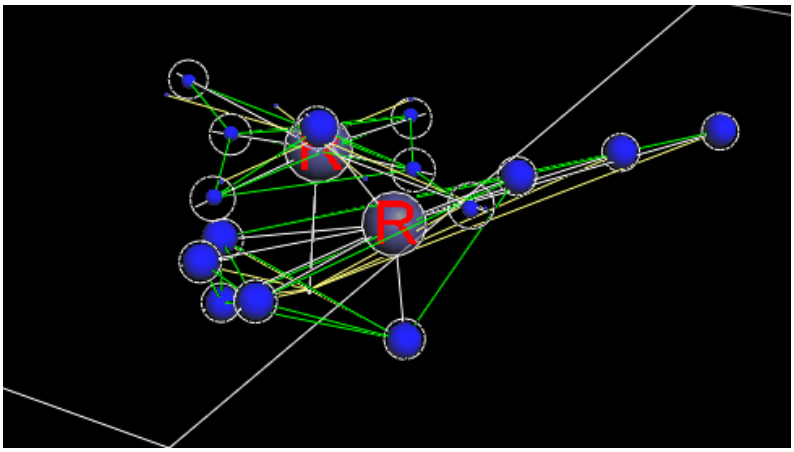

Figure 2.10: Virtual Marker-based Helicopter

Chapter 3

State Estimation

This section presents strategies that have been utilized or developed to estimate vehicle kinematic states using on-board sensor measurements. Model-based observers allow the reconstruction of state estimates given a subset of sensor readings so long as the observability condition is satisfied. If a model is not available, sensor estimates can be fused based on a complimentary filter approach designed to mitigate complimentary noise characteristics in the sensors.

A primary application for state estimation on MAVs is a reliable attitude estimate. Attitude feedback is generally the first level of control in a cascaded feedback loop design and therefore an accurate, high bandwidth attitude estimate is critical for efficient operation. Estimating attitude given various input measurements is a widely researched and well documented process. However, while there are several proven methods to accomplish the goal, selection of the ideal method depends upon available sensors and processing power, both of which tend to be limited on smaller vehicles.

3.1 Attitude Representation

The vehicle attitude, or orientation of the vehicle with respect to some inertially fixed coordinate system, can be represented in several varying ways. Two

of the most widely used methods, Euler angles and quaternions, each have unique benefits and drawbacks associated with their usage. Both representations have been employed in this research, with conversions occurring as necessary.

3.1.1 Euler Angles

Euler angles are a traditional representation of attitude in which an orientation in 3D space can be described by three successive rotations about orthogonal axes. A resulting orientation is sensitive to the order in which the rotations are performed, therefore the aerospace community has generally agreed that the order should be to first rotate by the yaw angle, then pitch, then finally the roll angle, to transform a vector from inertial to body coordinates. To transform from body to inertial coordinates, the procedure is reversed, from roll to yaw using negative angles. Given angles ψ, θ and ϕ, all three successive rotation matrices can be multiplied to produce a single transformation matrix, given as

$$R_{bi} = \begin{bmatrix} \cos\theta\cos\psi & \cos\theta\sin\psi & -\sin\theta \\ \sin\theta\sin\phi\cos\psi - \sin\psi\cos\phi & \sin\psi\sin\theta\sin\phi + \cos\psi\cos\phi & \sin\phi\cos\theta \\ \sin\theta\cos\phi\cos\psi + \sin\psi\sin\phi & \sin\psi\sin\theta\cos\phi - \cos\psi\sin\theta & \cos\phi\cos\theta \end{bmatrix}$$

Each successive rotation is described by an orthogonal rotation, and the product of three orthogonal matrices is also orthogonal, granting the above matrix the property that its inverse equals its transpose. From this the reverse transformation matrix is:

$$R_{ib} = R_{bi}^{-1} = R_{bi}^{T} \tag{3.1}$$

Given angular rates in body coordinates, the corresponding Euler rates depend upon the current Euler angles and can be calculated using[29]:

$$\dot{\phi} = p + \tan\theta(q\sin(\phi) + r\cos(\phi)) \tag{3.2}$$

$$\dot{\theta} = q\cos\phi - r\sin\phi \tag{3.3}$$

$$\dot{\psi} = \sec\theta(q\sin\phi + r\cos\phi) \tag{3.4}$$

One concern for some applications is that a singularity exists when the vehicle is pointing straight up, (Pitch = 180^{O}). An additional concern for embedded implementations is that each rotation involves several trigonometric calculations. This becomes a noticeable drawback when the processing resources are limited, such as they are with microcontrollers. For a typical 8-bit microcontroller, a $\sin(x)$ calculation can require over 200 times the number of clock cycles of a floating-point multiply operation, thus suggesting the investigation of the alternate, more computationally tractable, attitude representation in quaternions.

3.1.2 Quaternions

Quaternions were first described by William Rowan Hamilton as a class of hyper-complex numbers. They follow the rules of associativity but not commutativity[30]. A unique orientation is represented by a single rotation, D, about a single axis making angles of A, B, C with the inertial coordinate system. By using 4 parameters,

given as

$$e_0 = \cos\frac{D}{2} \tag{3.5}$$

$$e_1 = \cos A \sin\frac{D}{2} \tag{3.6}$$

$$e_2 = \cos B \sin\frac{D}{2} \tag{3.7}$$

$$e_3 = \cos C \sin\frac{D}{2} \tag{3.8}$$

the representation is over parameterized and therefore avoids singularities at all orientations. The transformation matrix from inertial to body coordinated is given as:

$$R_{ib} = \begin{bmatrix} e_o^2 + e_1^2 - e_2^2 - e_3^2 & 2(e_1e_2 + e_0e_3) & 2(e_1e_3 - e_0e_2) \\ 2(e_1e_2 - e_0e_3) & e_0^2 - e_1^2 + e_2^2 - e_3^2 & 2(e_2e_3 + e_0e_1) \\ 2(e_0e_2 + e_1e_3) & 2(e_2e_3 - e_0e_1) & e_0^2 - e_1^2 - e_2^2 + e_3^2 \end{bmatrix}.$$

Equating the corresponding elements of the above matrix with the aerospace-sequence Euler angle matrix, quaternions can be found in terms of Euler angles, given below[29].

$$e_0 = \cos\frac{\psi}{2}\cos\frac{\theta}{2}\cos\frac{\phi}{2} + \sin\frac{\psi}{2}\sin\frac{\theta}{2}\sin\frac{\phi}{2} \tag{3.9}$$

$$e_1 = \cos\frac{\psi}{2}\cos\frac{\theta}{2}\sin\frac{\phi}{2} - \sin\frac{\psi}{2}\sin\frac{\theta}{2}\cos\frac{\phi}{2} \tag{3.10}$$

$$e_2 = \cos\frac{\psi}{2}\sin\frac{\theta}{2}\cos\frac{\phi}{2} + \sin\frac{\psi}{2}\cos\frac{\theta}{2}\sin\frac{\phi}{2} \tag{3.11}$$

$$e_3 = -\cos\frac{\psi}{2}\sin\frac{\theta}{2}\sin\frac{\phi}{2} + \sin\frac{\psi}{2}\cos\frac{\theta}{2}\cos\frac{\phi}{2} \tag{3.12}$$

It can be seen from the above that using quaternions for attitude representation can greatly reduce presence of trigonometric evaluations, thus making them attractive for embedded applications. Quaternion rates can be computed from body rates using the following equations:

$$\dot{e}_0 = -\frac{1}{2}(e_1 p + e_2 q + e_3 r) \tag{3.13}$$

$$\dot{e}_1 = \frac{1}{2}(e_0 p + e_2 r + e_3 q) \tag{3.14}$$

$$\dot{e}_2 = \frac{1}{2}(e_0 q + e_3 p + e_1 r) \tag{3.15}$$

$$\dot{e}_3 = \frac{1}{2}(e_0 r + e_1 q + e_2 p) \tag{3.16}$$

where p, q, and r are the rates in the body-fixed coordinate system. For a given quaternion to appropriately represent a point on a four-dimensional hypersphere and therefore a particular orientation, the magnitude must be unity, i.e. $\sqrt{e_0^2 + e_1^2 + e_2^2 + e_3^2} = 1$. Numerical roundoff errors arising from quaternion operations can be checked by re-normalized after any operation. For their simpler computational complexity and lack of singularities, quaternions were chosen as the preferred method of attitude representation for this research whenever practical.

3.2 Attitude Estimation Techniques

States can be estimated using deterministic and statistical frameworks. Deterministic methods are required if only one source of data for a given state is available but demonstrate an added advantage of simplicity. Statistical methods are more suitable for combining measurements given multiple sources of the same state. Iner-

tial measurement in general is better suited to detecting attitude than translational motion, thus translational motion estimation using inertial measurement is unreliable unless additional translational sensor information is known, such as GPS(Global Positioning System) position and velocity data, or vision-based sensing.

Attitude estimation is the process of comparing known sensor information in the body coordinates and attempting to deduce the current attitude with respect to an inertial reference system. Most often, the known inertial reference coordinate system represents a dynamic equilibrium point for the body under observation. The known inertial reference used for a rotary-wing MAV is straight up with respect to gravity, with no off-axis pitch or roll angle. Sensor information is often available as a sensor triad where three orthogonal sensors detect the projection of measured quantity along each axis, such is the case with tri-axial magnetometers and tri-axial accelerometers. Given a multitude of such tri-axial vector measurements, usually pre-filtered, a classic solution for condensing many measurements down to a single rotation matrix was proposed by Wahba[31], where the individual contributions from each vector measurement are combined and weighted by a cost function. The fundamental requirement for Wahba's solution, as well as any attitude estimation technique, is the knowledge of at least two non-coincident vectors.

One additional method of estimating rotation angle that does not rely on observing external measurements in the body frame is the time integration of rotation rate. Rate gyro sensors are often available to provide a direct estimate of rate. Integrating rotation rate numerically, especially using MEMS-based rate gyros, will result in a relatively fast diverging error. This method however is reliable

for short term measurements, and can be very effectively combined with a method that directly measures attitude.

The TRIAD algorithm is a deterministic algorithm to find an attitude rotation matrix given two measured non-parallel vectors in the body frame. Assume $\vec{a}_b$ and $\vec{b}_b$ are two such given measurements in the body frame. The vectors are known to be $\vec{a}_i$ and $\vec{b}_i$ in the inertial frame. A rotation matrix can be constructed by finding a matrix R_{bi} such that $\vec{a}_b = R_{bi}\vec{a}_i$ and $\vec{b}_b = R_{bi}\vec{b}_i$. In general, due to errors in the measurements causing non-orthogonality of the vectors, the same rotation matrix R_{bi} will not fit both transformations. To estimate a single rotation matrix requires coercion of one or both measurement vectors.

3.2.1 Traditional TRIAD Algorithm

The traditional TRIAD algorithm [32] assumes one measurement is known with a greater accuracy than the other. In this case, finding R_{bi} involves the intermediate step of finding two matrices, R_{bt} and R_{ti} such that $R_{bi} = R_{bt}R_{it}^T$. The new intermediate frame, t, is defined as follows:

First, the more reliably known measurement vector, denoted as $\vec{a}_b$, is defined unchanged to be t_{1b}, given as equation 3.18. Likewise, $\vec{a}_i$, the known inertial component vector for the first measurement is defined as t_{1i}, as given in equation 3.19.

$$\widehat{t_1} = \widehat{a}_1 \tag{3.17}$$

$$t_{1b} = a_b \tag{3.18}$$

$$t_{1i} = a_i \tag{3.19}$$

Secondly, t_{2b} and $t2i$ are given as the normalized cross product between the two vectors in each frame. t_{2b} is given in equation 3.21, and t_{2i} is given in equation 3.22.

$$\widehat{t_2} = \widehat{a} \times \widehat{b} \tag{3.20}$$

$$t_{2b} = \frac{a_b \times b_b}{\|a_b \times b_b\|} \tag{3.21}$$

$$t_{2i} = \frac{a_i \times b_i}{\|a_i \times b_i\|} \tag{3.22}$$

The third step is to calculate the normalized cross product of $\widehat{t1}$ and $\widehat{t2}$ to find the third orthogonal vector, as given in equations 3.24 and 3.25. If $\vec{a}$ and $\vec{b}$ were originally exactly perpendicular, this third step would yield a vector collinear with $\vec{b}$. Otherwise, this operation will coerce $\vec{b}$ into an orthogonal representation with $\vec{a}$, which can be seen in figure 3.1.

$$\widehat{t_3} = \widehat{t_1} \times \widehat{t_2} \tag{3.23}$$

$$t_{3b} = \frac{t_{1b} \times t_{2b}}{\|t_{1b} \times t_{2b}\|} \tag{3.24}$$

$$t_{3i} = \frac{t_{1i} \times t_{2i}}{\|t_{1i} \times t_{2i}\|} \tag{3.25}$$

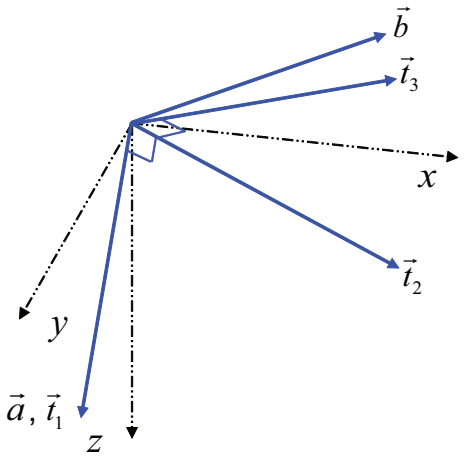

Figure 3.1: TRIAD Vectors

Finally, these calculated values for $\widehat{t_b}$ and $\widehat{t_i}$ are used to fill in the columns of R_{bt} and R_{it}, respectively, given as

$$R_{bt} = [t_{1b} \mid t_{2b} \mid t_{3b}] \tag{3.26}$$

$$R_{it} = [t_{1i} \mid t_{2i} \mid t_{3i}] \tag{3.27}$$

R_{bi} is given by

$$R_{bi} = R_{bt} R_{it}^T \tag{3.28}$$

3.2.2 A Modified TRIAD Algorithm

A modified TRIAD algorithm[33] exists that is not sensitive to which vector is chosen as the primary. Given $\vec{a}$ and $\vec{b}$ as above, in both the inertial and body coordinates, unit vectors can be found as

$$\widehat{i}_{b,i} = \frac{\vec{a}_{b,i} + \vec{b}_{b,i}}{\|\vec{a}_{b,i} + \vec{b}_{b,i}\|} \tag{3.29}$$

$$\widehat{j}_{b,i} = \frac{\widehat{i}_{b,i} \times (\vec{a}_{b,i} - \vec{b}_{b,i})}{\|\widehat{i}_{b,i} \times (\vec{a}_{b,i} - \vec{b}_{b,i})\|} \tag{3.30}$$

$$\widehat{k}_{b,i} = \widehat{i}_{b,i} \times \widehat{j}_{b,i} \tag{3.31}$$

The matrices R_{bt} and R_{it}, described above can be calculated by filling in the columns with the calculated $\widehat{i}, \widehat{j}$ and $\widehat{k}$ vectors, as below,

$$R_{bt} = [\widehat{i}_b \mid \widehat{j}_b \mid \widehat{k}_b] \tag{3.32}$$

$$R_{it} = [\widehat{i}_i \mid \widehat{j}_i \mid \widehat{k}_i] \tag{3.33}$$

3.2.3 Complementary Filter

The complementary filter provides a simple method for combining measurements with complementary noise characteristics. This filter is an ideal method for combining measurements from a numerically integrated rate gyro signal and accelerometer/magnetometer direct attitude sensors. Integrating rate gyro signals numerically provides a good estimate of angle for short term periods, but integration drift causes inevitable divergence. The direct measurements of attitude provided by the accelerometers and magnetometer are sufficient only for long term angular estimate, as they are corrupted by translational acceleration and electro-magnetic interference, respectively. The accelerometer/magnetometer estimate provides the low-pass filtered input to the complementary filter, while the integrated rate gyro

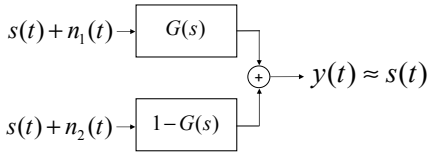

Figure 3.2: Complementary Filter

signals provide the high-passed input signal.

This implementation is shown in figure 3.2, where $G(s)$ corresponds to a low-pass filter and $(1 - G(s))$ is the complementary high pass filter. In this figure, $n_1(t)$ is noise associated with the accelerometer measurements and $n_2(t)$ is associated with the integrated rate gyro measurements. The signal $y(t)$ is the resulting improved attitude estimate.

3.2.4 Quaternion-Based Complimentary Filter for Attitude Estimation

A useful quaternion-based complementary filter was developed by Bachmann, et al.[1] for estimating limb orientation for virtual reality applications. The flow chart for this algorithm is given in figure 3.3. The algorithm is initialized with some arbitrary $\widehat{q}$. The direct accelerometer and magnetometer readings are concatenated into one vector, known as the *measurement vector*, y_o. The *computed measurement vector*, $y(\widehat{q})$, is found from the initialized quaternion using $\widehat{q}^{-1}m\widehat{q}$ and $\widehat{q}^{-1}n\widehat{q}$, where m and n are 4x1 pure quaternions defining the sensor outputs in inertial coordinates. The gravity vector in inertial coordinates, m, is given as $[0\ 0\ 0\ 1]^T$, while the magnetic field vector, n must be known for a given location and is given as $[0\ n_x\ n_y\ n_z]^T$.

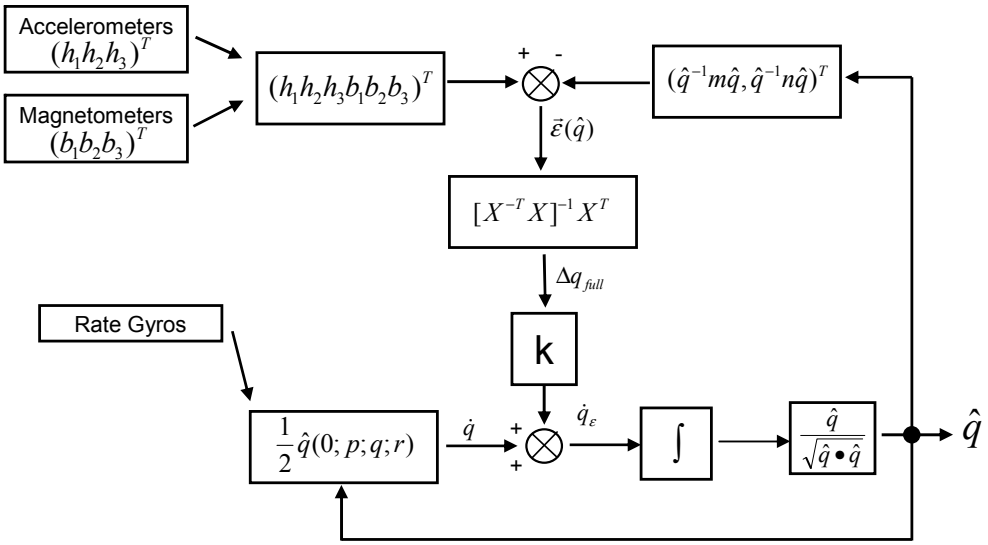

Figure 3.3: Quaternion-based Complementary Filter[1]

The error between the *computed measurement vector* and the *measurement vector* is given as $\epsilon(\widehat{q})$. The matrix X is given by $X_{ij} = \left[\frac{\partial y_i}{\partial q_j}\right]$. The error quaternion rate $\dot{q_\epsilon}$ is added to the quaternion rate given by the rate gyros, and then integrated in time. The algorithm utilizes Gauss-Newton iterations calculated from the direct attitude measurement to correct the drift inherent to the numerical integration of the rate gyro signals. The resulting quaternion $\widehat{q}$ is more accurate compared to the estimates obtained by either the accelerometer/magnetometer estimate or the integrated rate gyro estimate would be alone. The feedback nature of the algorithm allows one to vary the confidence in the direct attitude estimate with the confidence in the integrated rate gyro estimate simply by varying the gain parameter k, where $0 < k < 1$.

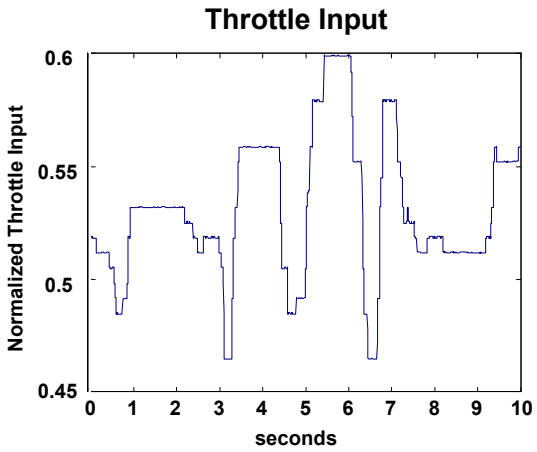

Figure 3.4: Time History of Throttle Input

3.3 Rotor RPM Estimation from Gyro Signal Frequency Analysis

The E-sky Honeybee electric helicopter is a fixed pitch helicopter that relies on variable rotor RPM to generate variable thrust, while larger helicopters generally utilize a governor to ensure rotor speed stays constant, and vary thrust via collective. A step change in throttle will not result in an instantaneous change in rotations per minute(RPM), due to the blade and motor inertia. Furthermore, the mapping from throttle position to steady-state rotor rotation rate is unknown.

Knowledge of the actual rotor RPM would help to give a better estimate of the current thrust produced rather than the throttle input. Embedded in the rate gyro signals is the expected two-per-rev high frequency oscillation associated with the rotor aerodynamics. The actual rotor RPM can be estimated without directly monitoring the RPM of the rotor. Using only additional processing power, a short-time Fourier transform analysis can extract the rotor RPM which is encoded in the high frequency content of the gyro rate signal.

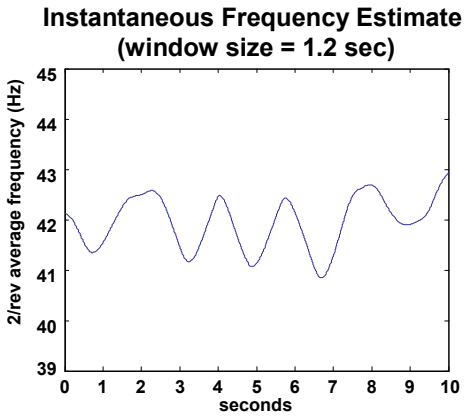

Figure 3.5: Short-Time Fourier Transform of Gyro Signal

Using the open source WaveLab[34] time-frequency analysis package, a Short-Time Fourier Transform(STFT) was applied to the post-flight unfiltered rate gyro data. The average frequency of the high passed STFT was computed for each time step. The lower frequencies were excluded to avoid corrupting the average frequency estimate with the angular rates due to the rigid body dynamics. The resulting function provided a good indication of the current rotor RPM. The actual rotor RPM can be obtained by simply dividing the 2/rev frequency estimate by 2. It was found from this analysis that the typical main rotor speed for hover was in the range of 20-21 Hz.

Figure 3.4 shows a typical variation in throttle during flight, while figure 3.5 shows the variation in 2/rev frequency of the main rotor for the same time segment. The slower, lagging response of the rotor RPM can be observed.

3.4 On-board Complimentary Filter for Heave Velocity Estimation

To implement on-board height control, a sonar range finder was utilized with the z-axis accelerometer present on the vehicle. While given altitude and vertical acceleration however, there is no direct velocity estimate. To obtain this, a complimentary filter concept was utilized that was presented in [35]. The resulting velocity estimate is the sum of filtered versions of the sonar and accelerometer output. A particular frequency cutoff control allows a tradeoff adjustment between confidence in the absolute position estimate and the acceleration measurement.

3.5 Translational Velocity Estimation from Ventral Optic Flow

Measurements of translational velocity for are not typically available at a high update rate, particularly when relying upon GPS or pressure sensors. A simple implementation of optic flow that would permit a high bandwidth estimate of translational velocity utilizes the sensor combination in figure 2.3.

A diagram of how the optic flow sensor and sonar altimeter can be used for translational velocity estimation is shown in figure 3.6. In the figure, an MAV is shown translating with some velocity, V, a rotation rate, ω, and at a height, h, from the ground. The one dimensional optic flow, ξ, assuming the average optic flow in a small field of view, induced by this sensor arrangement is given as

$$\xi = -\omega + V/h \tag{3.34}$$

Referring to equation 3.34, assuming negligible pitch or roll rotation, the ob-

served optic flow field is proportional to the ratio of V/h. The velocity field is not uniform for a given fixed translational speed, however. Assuming the sensor is imaging a flat ground surface, the distance to each point in the visual field varies. The distance, h, denotes the distance along the optical axis to the ground. It should be emphasized that in this configuration the vehicle would only sense and avoid ventrally located obstacles.

Optic flow itself is insufficient to determine translational velocity, particularly if the depth of the imaged environment is unknown. However, by operating the vehicle only over flat terrain, or terrain that can be regarded as flat from a distance, such as a field with short grass, the translational velocity can be estimated by knowing the distance to one point in the field of view, along with the self rotation. In this case, an ultrasonic ranging sensor, shown in figure 2.3 was used to provide a direct measurement of distance to the ground (along the body coordinate z axis in figure 3.6), and an on-board rate gyro provided rate information.

This is sufficient to constrain the solutions of the optic flow equation to then extract translational velocity. The optic flow due to rotation was first separated using the rate gyro data. This separation is shown in figure 3.7. The the optic flow due to translation is scaled by the current sonar height estimate, as shown in figure 3.8. Noise in the optic flow and the structured uncertainty that arises when attempting to subtract off the rotation rate limits the transient accuracy of the estimate, and hence limits the overall gain that can be applied to a feedback loop that utilizes the estimate.

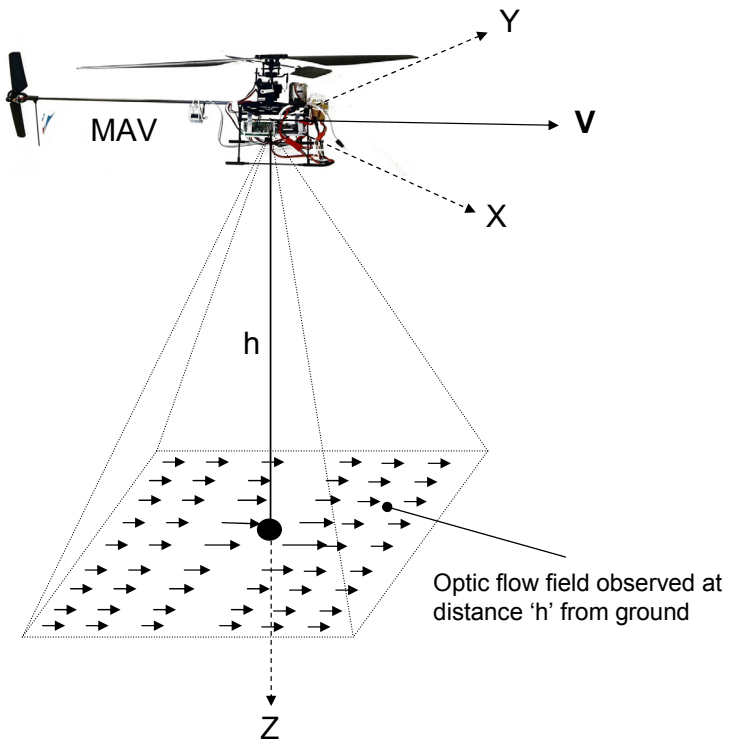

Figure 3.6: Optic Flow Sensor and Sonar Arrangement for Translational Velocity Estimation

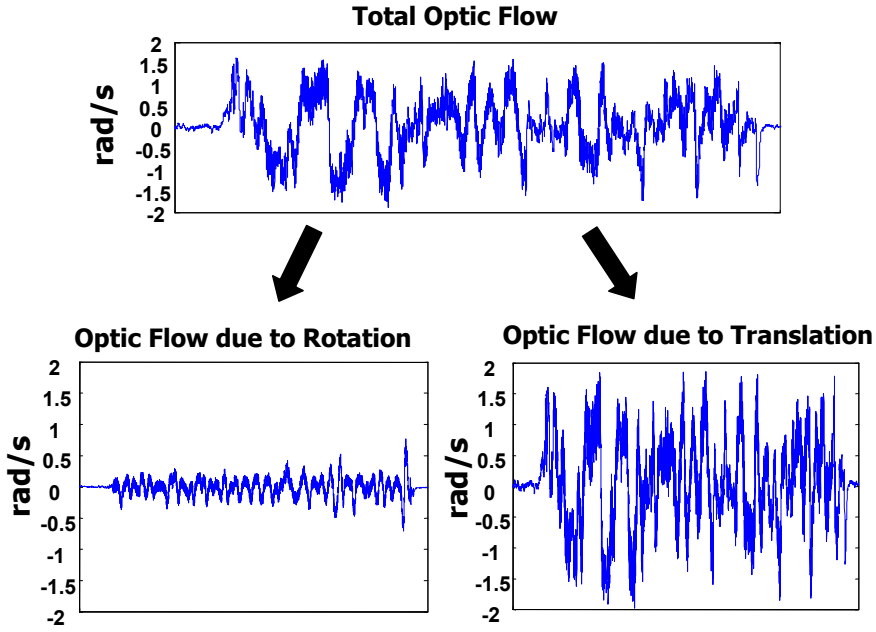

Figure 3.7: Optic Flow Separated into Rotational and Translational Optic Flow.

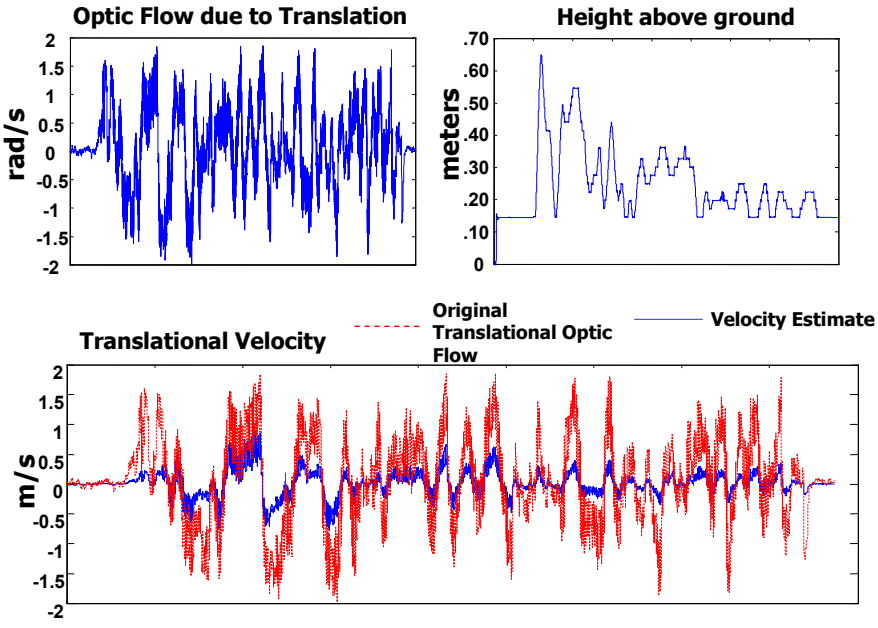

Figure 3.8: Velocity Estimation from Optic Flow and Sonar.

3.6 Off-board State Estimation

The Vicon obtained estimates for position and orientation are directly measured and exceptionally low noise compared to those that can be obtained from on-board sensors. The position and orientation noise variances were estimated by recording data while the vehicle remained stationary. The variance of the noise for the position was estimated to be 3.54×10^{-3} meters, the variance corresponding to the orientation was estimated to be 2×10^{-5} rad.

The inertial-frame fixed positions can be numerically differentiated to provide velocities in the inertial frame. The inertial velocity can be calculated as:

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{pmatrix} = \frac{d}{dt} \begin{pmatrix} x \\ y \\ z \end{pmatrix} \tag{3.35}$$

Body fixed velocities can be directly computed using the direction cosines matrix representation of the orientation estimate, R_{bi} and the inertial velocities as seen in equation 3.36. The body fixed velocity of the vehicle's center of gravity (CG) is given by u,v, and w for longitudinal velocity, lateral velocity, and heave velocity, respectively. The body-fixed velocities have an estimated noise variance of 6.4×10^{-2} m/s.

$$\begin{pmatrix} u \\ v \\ w \end{pmatrix} = R_{bi} \begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{pmatrix} \tag{3.36}$$

Chapter 4

System Identification of an Electric Micro-Helicopter

This chapter focuses on the development of micro air vehicle dynamic modeling with particular application to the single main rotor Honeybee vehicle. A typical approach to autonomous operation of a rotary wing MAV requires sensing, processing, feedback control, and state estimation. The feedback control and state estimation strategies can employ a dynamic model of the vehicle to aid in the calculation of the current states and control inputs required to regulate the vehicle. Typically, a linearized model is sufficient to describe the dynamics close to some desired flight condition.

A linearized model can be obtained from the full nonlinear equations of motion for the vehicle using first principles derivations. Considering the full nonlinear model of a helicopter, this approach may yield a model of unknown accuracy due to often drastic assumptions required to simplify the problem. Additionally, it might be difficult to measure or estimate the physical characteristics of the vehicle. Micro air vehicles below a certain scale operate in a range of Reynolds numbers where the aerodynamics do not behave as predicted for larger vehicles[36], therefore aerodynamic predictions from first principles can be unreliable.

For these aforementioned reasons, the following sections present the identification of a linearized model using flight data as an alternative to first principles vehicle

modeling. The identification methodology utilized frequency responses to estimate transfer functions between inputs and outputs. The frequency response method was based on prior work of Tischler[37] and Mettler[38] for larger vehicles, including the identification of BO-105 dynamics and the Yamaha R-max dynamics. Frequency response analysis was performed using the Comprehensive Identification of FRequency Responses (CIFER®) software developed at Army Aeroflightdynamics Directorate (AFDD) located at Moffett Field, CA.

A state space model in the form of equations 4.1 and 4.2 was then constructed from the computed transfer functions, where $\mathbf{x}$ is the state vector, $\underline{\mu}$ is the control vector, $\mathbf{A}$ is a linear, time invariant system matrix, $\mathbf{B}$ is a matrix of constant control sensitivity derivatives, and $\mathbf{C}$ is the output equation matrix. The state space model representation was assembled to match that of a typical linearized helicopter model[39], which allowed a direct comparison of stability derivatives with other larger helicopters.

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\underline{\mu} \tag{4.1}$$

$$\mathbf{y} = \mathbf{C}\mathbf{x} \tag{4.2}$$

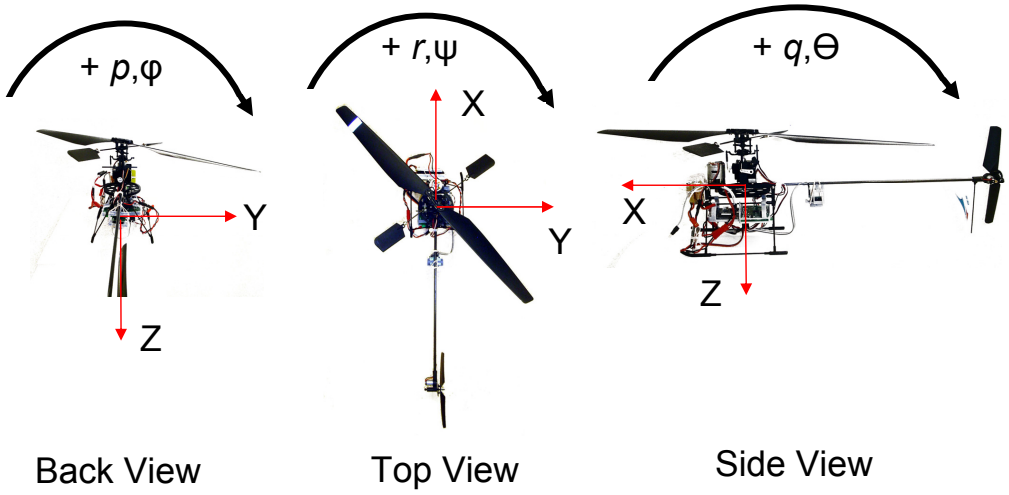

Figure 4.1: Helicopter Coordinate System

4.1 Experimental Setup

4.1.1 Flight Test Vehicle

The flight vehicle is the commercially available E-sky brand Hobby Helicopter, shown in figure 1.2. The main rotor is 50.5 cm in diameter, and the tail rotor is 14.5 cm in diameter. With the original components integrated, the vehicle weighed 288 grams. The original vehicle electronics were replaced with the custom designed avionics package. Additionally, the original NiMh battery was replaced with two Thunderpower® Lithium Polymer batteries, one for the vehicle and one for the electronics. The vehicle modifications resulted in a final gross vehicle weight of 390 grams. The chosen body-fixed coordinate system is shown in figure 4.1 and follows the standard flight vehicle convention with the X-axis positive forward, Z positive downward, and Y positive starboard. The avionics board provides the cen-

tral interface between the sensors, actuators, and communications. The component connection diagram is shown in figure 4.2. The servos and speed controllers are standard RC components.

The flight test vehicle configuration is similar to a standard helicopter and as such, utilizes a single main rotor for thrust, a single tail rotor for counteracting main rotor torque and yaw control, and a swashplate mechanism for cyclic control. The cyclic control on the test flight vehicle is implemented via a Hiller style flybar assembly that influences the main rotor flapping angles. The rotorhead assembly can be seen in figure 4.3. The flybar flapping completely determines the angle of attack of the main blades and there is no direct linkage between the servos and the main rotor angle of attack. The main blades are molded plastic material and relatively stiff. The blades are firmly cantilevered at the rotor hub and as such, constitute a hingeless rotor design which maximizes the induced moments about the rotor hub. Small weights attached to the paddles provide additional inertia to slow the rotor response. The thrust generated by the tail and main rotors is dependent on variable speed and not collective control as is common with larger helicopters.

4.2 Available Inputs and Outputs for Identification

4.2.1 Vehicle Inputs

Nothing was presumed to be known about what forces or deflection angles were generated given a change in the actuator, therefore all control inputs simply normalized. The cyclic commands are given by μ_{lat} and μ_{lon} for lateral or roll input

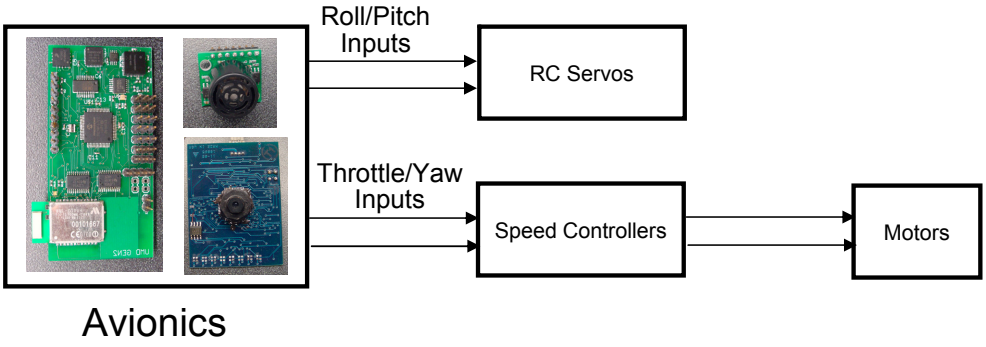

Figure 4.2: Connection Diagram.

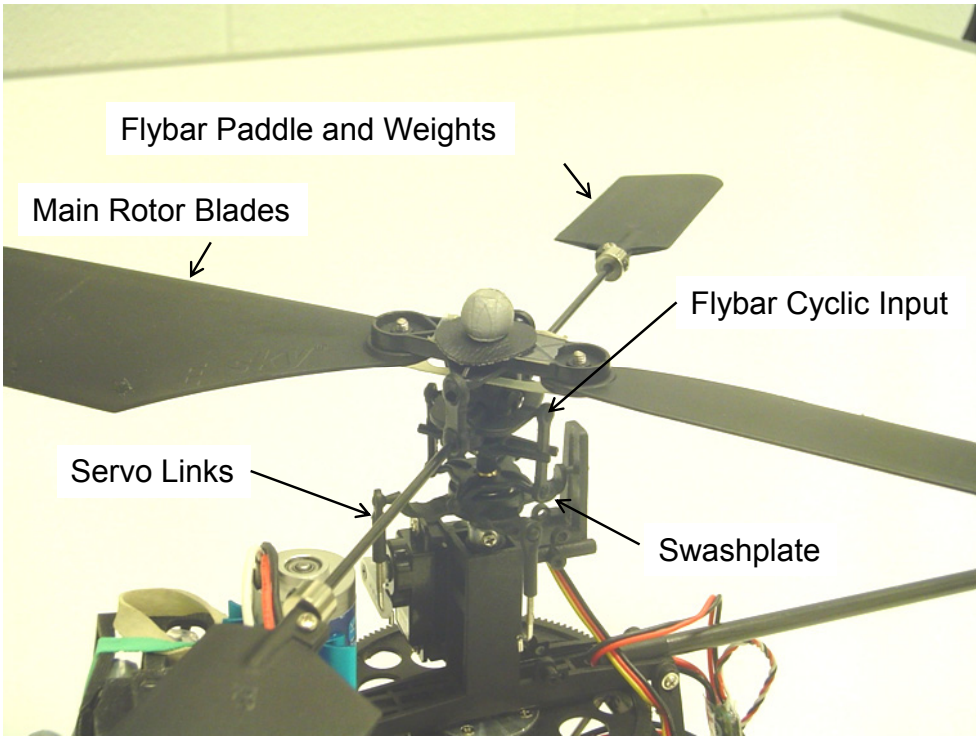

Figure 4.3: Flight Vehicle Rotorhead Mechanism

and longitudinal or pitch input, respectively. These inputs were normalized such that $\mu_{lat} \in [-1, 1]$ and $\mu_{lon} \in [-1, 1]$, while the tail input, μ_y, and main rotor input, μ_t, were normalized such that $\mu_y \in [0, 1]$ and $\mu_t \in [0, 1]$

4.2.2 Available Kinematic Outputs

The inertial position and orientation and body-fixed kinematics were available using the Vicon visual tracking system. The rotor speed was estimated from the numerical derivative of the rotor Euler yaw angle estimate. Observation of the rotor rotational velocity was useful to the identification of the heave dynamics for this vehicle, as the main rotor thrust is affected directly using the main rotor speed rather than the more traditional collective blade pitch control. The roll and pitch orientation of the rotor was not considered during the analysis. The inertial positions were also not useful for the subsequent analysis. The kinematic state vector made available to the identification process is shown in equation 4.3.

$$y = \begin{bmatrix} p & q & r & \phi & \theta & \psi & u & v & w & \Omega_{mr} \end{bmatrix}^T \tag{4.3}$$

4.3 Flight Test Data

Piloted maneuvers were performed that excited the vehicle over a range of frequencies. All flight testing was performed within the capture volume of the visual tracking system. Low frequency inputs generally lead to large translations, hence the physical capture area limited the lowest frequency inputs possible. Additionally, the

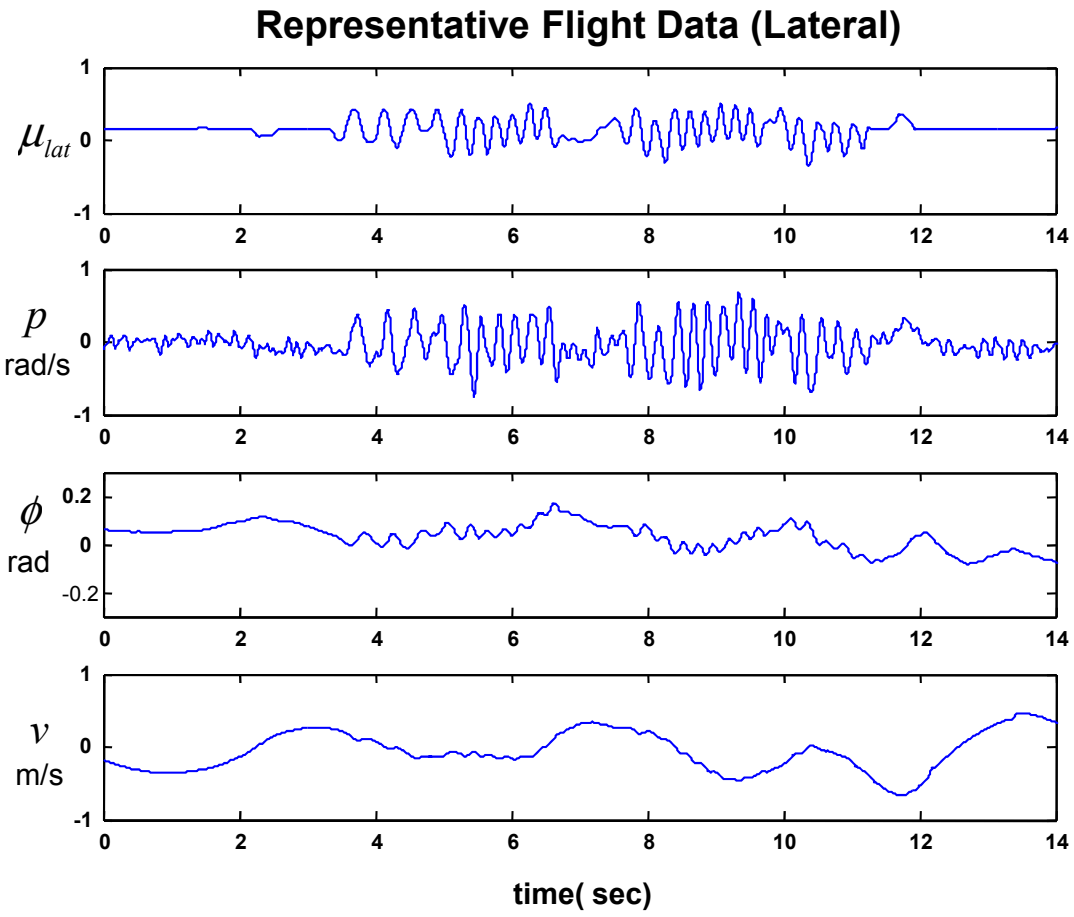

Figure 4.4: Representative Lateral Flight Data Maneuver.

inherent high responsiveness of the vehicle made long, unbroken frequency sweeps difficult. Flight data sufficient for identification, was possible via the concatenation of a series of 'doublet' style inputs and short chirp-like frequency bursts between periods of returning the vehicle to the center of the testing area. A typical portion from a recorded data set is shown in Figures 4.4 and 4.5.

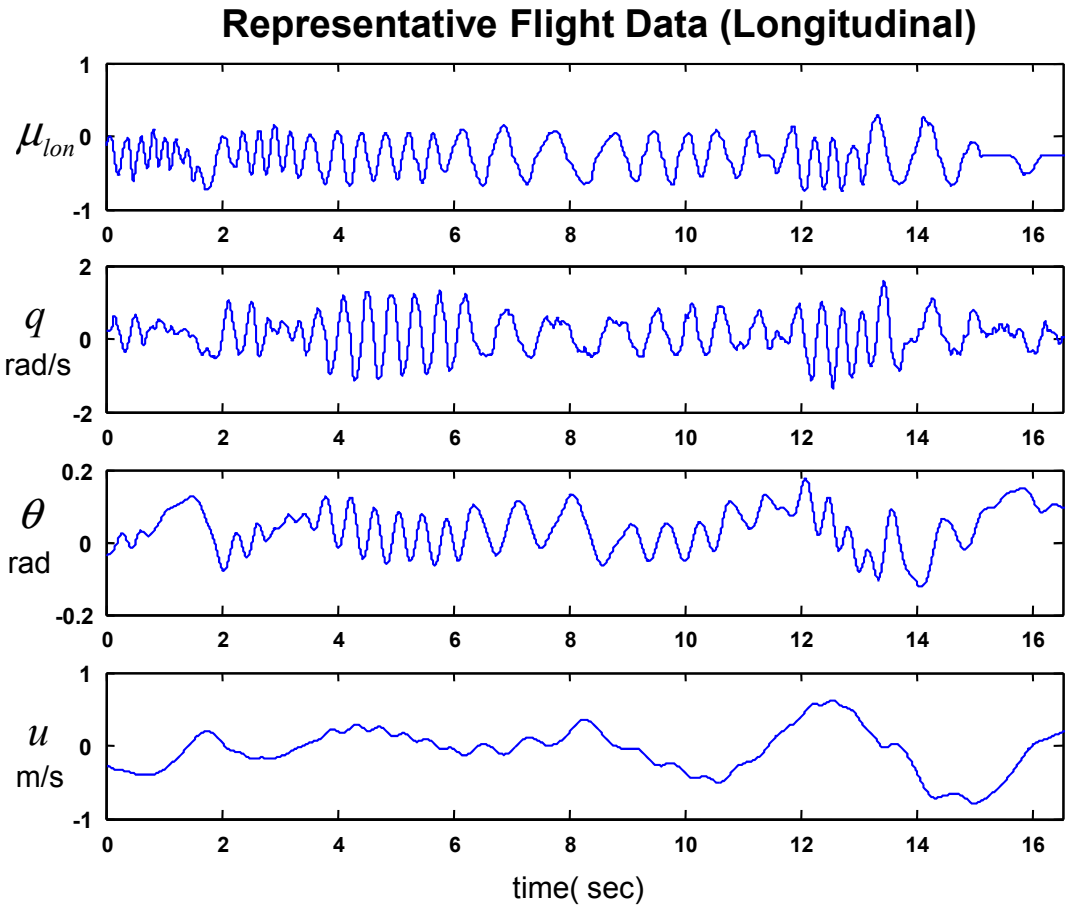

Figure 4.5: Representative Longitudinal Flight Data Maneuver.

4.4 Model Structure

The structure of the vehicle state space model was based primarily on prior successful work on the identification of helicopter models[40, 41, 39]. Some characteristics unique to the test vehicle required a basic model derivation of the linear dynamics from first principles. While the result was a full state space model of all relevant dynamics, the identification proceeded by analyzing decoupled subsystems independently and concatenating to build the full model.

4.4.1 Cyclic Inputs to p and q Body Rates

The lateral and longitudinal cyclic inputs influence the flapping angle of the rotor disk, thus altering the moments about the vehicle. Helicopters typically exhibit a non-negligible coupling between the cyclic inputs and the induced off-axis moments. The cyclic actuation mechanism of the test flight vehicle is typical of current low cost micro helicopters, however, it is somewhat atypical compared to those of larger vehicles. There is no direct linkage between the cyclic inputs and the blade pitch angle. Instead, a weighted flybar with Hiller-style paddles governs the angle of attack of the main blades, and the pilot cyclic inputs only affect the cyclic angle of attack of the flybar. Most larger unmanned helicopters also utilize a flybar for stability, however they also generally include some direct mechanical linkage between the pilot inputs and the main rotor blade pitch, often referred to as a Bell-Hiller mixer[42]. One benefit for using the flybar only as the input to the main rotor blades is that no forces are transmitted back to the servos, thus reducing

power required for actuation. Additionally, the response of the rotor to inputs and gusts is slower than if the servos directly actuated the rotor blades.

Several general assumptions were made which were consistent with similar prior work[39, 42]. The hingeless main rotor blades can be modeled as hinged rotors with an effective offset hinge and a root spring. The blades were assumed to exhibit no lead or lag motion, and the inflow dynamics were neglected, assuming the inflow angle to be small.

$$\begin{Bmatrix} \dot{b} \\ \dot{p} \\ \dot{a} \\ \dot{q} \end{Bmatrix} = \begin{bmatrix} -\frac{1}{\tau_f} & -1 & \frac{B_a}{\tau_f} & 0 \\ L_b & 0 & 0 & 0 \\ \frac{A_b}{\tau_f} & 0 & -\frac{1}{\tau_f} & -1 \\ 0 & 0 & M_a & 0 \end{bmatrix} \begin{Bmatrix} b \\ p \\ a \\ q \end{Bmatrix} + \begin{bmatrix} \frac{B_{lat}}{\tau_f} & \frac{B_{lon}}{\tau_f} \\ 0 & 0 \\ \frac{A_{lat}}{\tau_f} & \frac{A_{lon}}{\tau_f} \\ 0 & 0 \end{bmatrix} \begin{Bmatrix} \mu_{lat} \\ \mu_{lon} \end{Bmatrix} \tag{4.4}$$

The model structure used in this analysis is the *hybrid model* structure, initially proposed by Tischler and Cauffman[40]. This formulation, as opposed to the simpler *quasi-steady* model, accurately captures the coupled rotor/fuselage dynamics and is applicable to a wide range of design configurations. Mettler[39] utilized this formulation for the identification of the Yamaha Rmax dynamics and provided further validation that the *hybrid model* formulation was sufficient to capture the relevant dynamics while the *quasi-steady* model proved sufficient only for low frequencies.

The hybrid model structure, in state space form, is given in equation 4.4. As formulated, this model does not account explicitly for the dynamics of the flybar and instead considers the rotor and flybar to be a lumped system, with an effective time constant given as τ_f. The flapping angle states, b and a, describe the flapping angles

of the lumped rotor/flybar system. The fuselage roll and pitch rates, respectively are given by p and q. B_a and A_b capture the off-axis dynamics between flapping states, and L_b and M_a are the rotor moment derivatives, capturing the roll and pitch moments induced by the rotor. The primary control derivatives are B_{lat} and A_{lon} which represent the rate of change in cyclic angle to the swashplate given changes in the normalized pilot inputs. B_{lon} and A_{lat} are included to capture off-axis couplings. The control inputs are the normalized cyclic inputs to the flybar, μ_{lat} and μ_{lon}.

4.4.2 Yaw Input to Yaw Rate

Yaw rate is affected by the torque generated as a function of main rotor torque and torque generated by the tail rotor. The torque generated due to the main rotor is given in equation 4.5.

$$T_{mainrotor} = \frac{1}{6}\rho R_{mr}^3 \Omega_{mr}^2 c_{mr} C_{d_{mr}} \times \#\text{ of blades} \tag{4.5}$$

where ρ is the air density, R_{mr} is the radius of the rotor, Ω_{mr} is the rotor speed, c_{mr} is the blade chord, and C_{dmr} is the drag coefficient of the main rotor. The torque about the vehicle center of gravity (CG) can be given as:

$$T_{tail} = \frac{1}{6}\rho R_{tail}^3 \Omega_{tail}^2 c_{tail} C_{l\alpha_{tail}} * (\#\text{ of tail blades}) \tag{4.6}$$

where $C_{l\alpha}$ is the tail lift coefficient and c_{tail} is the tail rotor chord. Given these inputs, a nonlinear equation of motion for the yaw degree of freedom is given in equation 4.7 where N_r is the aerodynamic drag-induced torque about the vehicle

CG ahd l_{tail} is the length of the tail moment arm.

$$\dot{r} = -N_r r + l_{tail}\frac{1}{3}\rho R_{tail}^3 \Omega_{tail}^2 c_{tail} C_{l\alpha_{tail}} - \frac{1}{3}\rho R_{mr}^3 \Omega_{mr}^2 c_{mr} C_{d_{mr}} \tag{4.7}$$

Note that Ω_{tail} is the rotational speed of the tail rotor and Ω_{mr} is the main rotor rotational speed, both of which are the independent control inputs. A linearized version of this equation about the hover condition can proceed by assuming $\Omega_{mr} = \Omega_1$ and $\Omega_{tail} = \Omega_2$, where Ω_1 and Ω_2 are the nominal speeds of the main and tail rotors to affect a net zero torque in the hovering flight condition. Assuming the throttle changes very little and slowly about hover and any vertical heave motion is slow, the following linearized equation, equation 4.8, can be used.

$$\dot{r} = -N_r r + N_{\mu_y}\mu_y \tag{4.8}$$

The yaw input is represented by μ_y, and the output, r, is the yaw rate. An open loop mixing does exist where the yaw input increases linearly with the throttle input. This mixing was preset and is known exactly, therefore can be taken into account during the identification. The identification proceeded assuming these first order dynamics.

The identification of the transfer function from the yaw input to the yaw rate was complicated by the presence of the feedback to make the vehicle pilotable. To obtain the yaw transfer function, the vehicle was fixed to a freely rotating test stand that allowed motion only in the yaw degree of freedom. Beginning from the nominal trim, a computerized frequency sweep was given to the vehicle with no feedback and

the response was recorded. The rotational inertia and friction of the test stand was assumed to be negligible compared to the vehicle dynamics and this test sweep was included in the analysis. No coherence was observed between the rotational rate in flight and the heave velocity therefore the N_w stability derivative, which describes the vertical acceleration induced by yaw motion, was not identified and fixed to zero.

4.4.3 Input to Body Orientation Transfer Function Estimation

The angular velocity of the vehicle in body coordinates can be transformed into the angular velocity expressed in inertial coordinates, the time derivative of the Euler angles, via the following transformation:

$$\begin{Bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{Bmatrix} = \begin{bmatrix} 1 & \sin\phi\,\tan\theta & \cos\phi\,\tan\theta \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi/\cos\theta & \cos\phi/\cos\theta \end{bmatrix} \begin{Bmatrix} p \\ q \\ r \end{Bmatrix} \tag{4.9}$$

Considering the case where the vehicle is linearized about the hover condition, the body and inertial rate vectors are approximately equal, $\omega_b \approx \omega_i$, therefore $\dot{\theta} \approx \omega_{by}$, $\dot{\phi} \approx \omega_{bx}$, and $\dot{\psi} \approx \omega_{bz}$ in the linearized model. This implies that the transfer functions for the cyclic inputs to the roll and pitch angular orientation should simply be the transfer functions from the cyclic inputs to the body rates multiplied by $\frac{1}{s}$.

4.4.4 Orientation to Translational Velocity Transfer Function Estimation

Due to a lack of excitation and high control coupling at low frequencies, direct estimation of translational velocity damping from the lateral and longitudinal inputs may be difficult. Instead, the lateral damping stability derivatives can be estimated as first order poles in transfer functions from the roll and pitch angular orientations to the lateral and longitudinal translational velocities. This method, using the simplified first order equations in equation 4.10, was employed to identify the Y_v and X_u damping derivatives.

$$\begin{aligned} \dot{v} &= Y_v v + g\phi \\ \dot{u} &= X_u u + g\theta \end{aligned} \tag{4.10}$$

The corresponding transfer functions from the orientation to translational velocities are given as:

$$\begin{aligned} \frac{v}{\phi}(s) &= \frac{g}{s - Y_v} \\ \frac{u}{\theta}(s) &= \frac{g}{s - X_u} \end{aligned} \tag{4.11}$$

Once these derivatives were identified independently, they were then included in the full state space model.

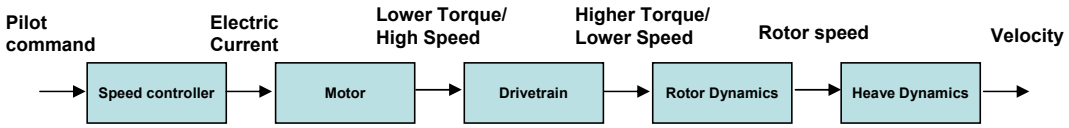

Figure 4.6: Schematic of Thrust Dynamics for Electrically Driven Helicopters

4.4.5 Throttle to Heave Velocity

For the configuration of most manned and unmanned helicopters, the main rotor and tail rotor are slaved together, either using a gear-based drivetrain or a flexible belt drive, and thrust is generated by varying the collective pitch of the blades. The tail and main rotor forces for the test vehicle in this research, however, are independently controlled via the rotational speed of electric motors, with no collective control. Figure 4.6 shows the drivetrain of the test vehicle, composed of RC hobby components. Each component of the drive system contains internal dynamics and contributes small nonlinear effects, but these effects are negligible in comparison to the primary inertial and aerodynamic forces and moments, therefore in the interest of keeping the model simple, it was assumed that the pilot's command generates some unknown torque that is transmitted to the rotor with some small delay Δ_{th}. The transfer function from throttle input to rotor speed was modeled as a simple first order response, seen in equation 4.12, accounting for the rotor inertia and drag on the spinning blades.

$$\dot{\Omega}_{mr} = -T_{\Omega_{mr}}\Omega_{mr} + T_{\mu_t}\mu_t \tag{4.12}$$

Given a rotor speed, Ω_{mr}, the lift generated by the main rotor can be given

as:

$$T_{mr} = \frac{1}{6}\rho R^3 \Omega^2 c_{mr} C_{l_\alpha} \times \#\text{ of blades} \tag{4.13}$$

From this equation is it apparent that varying the rotor speed to change lift is a nonlinear relationship. If the assumption is made that the thrust varies very little about the operating condition, then a linearized first order model of Ω_{mr} to w can be estimated.

A linear transfer function between the rotor speed and heave velocity was estimated in the attempt to capture the heave damping. The equation describing the linearized heave dynamics is given in equation 4.14, where Z_w describes the damping with vertical velocity and $Z_{\Omega_{mr}}$ is the heave acceleration in response to a change in rotor speed.

$$\dot{w} = -Z_w w + Z_{\Omega_{mr}} \Omega_{mr} \tag{4.14}$$

Equations 4.14 and 4.12 can be collected into the state space model describing the heave dynamics. This model, used for the identification process, is given in equation 4.15.

$$\begin{Bmatrix} \dot{w} \\ \dot{\Omega}_{mr} \end{Bmatrix} = \begin{bmatrix} Z_w & Z_{\Omega_{mr}} \\ 0 & T_{\Omega_{mr}} \end{bmatrix} \begin{Bmatrix} w \\ \Omega_{mr} \end{Bmatrix} + \begin{bmatrix} 0 \\ T_{\mu_t} \end{bmatrix} \mu_t \tag{4.15}$$

4.4.6 L_v and M_u Derivatives

The L_v and M_u stability derivatives are the source of the unstable modes inherent in rotorcraft. These derivatives typically induce very low frequency oscillatory modes. These derivatives were first fixed to zero, until good estimates were obtained for the derivatives in equation 4.4. Once the parameters in equation 4.4 were identified with good confidence, and the the lateral and longitudinal speed derivatives were identified and fixed, these derivatives freed along with the other parameters and allowed to converge.

4.4.7 Full State Space Model

The full assembled state space model is given in equation 4.16. In general, there may exist many specific state space models that would exhibit the frequency responses identified during testing, however by assembling the state space structure based on the physics of the vehicle, the resulting degrees of freedom in the identification are reduced, as is evident by the large number of zero values in equation 4.16. By fitting the constrained model to the identified transfer functions, the resulting identified stability derivatives have some physical significance that can provide useful insight into the vehicle dynamics.

The control inputs are primed in equation 4.16 indicating that these inputs include the pure delays given in table 4.1. The $\Delta t_{\mu_{cyclic}}$ term is the pure delay in seconds for both cyclic actuators. The terms Δt_{μ_t} and Δt_{μ_y} specify the delay for the throttle and yaw motors, respectively.

$$\begin{Bmatrix} \dot{b} \\ \dot{p} \\ \dot{a} \\ \dot{q} \\ \dot{\phi} \\ \dot{\theta} \\ \dot{r} \\ \dot{w} \\ \dot{v} \\ \dot{u} \\ \dot{\Omega}_{mr} \end{Bmatrix} = \begin{bmatrix} -\frac{1}{\tau_f} & -1 & \frac{B_a}{\tau_f} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ L_b & 0 & 0 & 0 & 0 & 0 & 0 & 0 & L_v & 0 & 0 \\ \frac{A_b}{\tau_f} & 0 & -\frac{1}{\tau_f} & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & M_a & 0 & 0 & 0 & 0 & 0 & 0 & M_u & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & N_r & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & Z_w & 0 & 0 & Z_{\Omega_{mr}} \\ 0 & 0 & 0 & 0 & g & 0 & 0 & 0 & Y_v & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -g & 0 & 0 & 0 & X_u & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & T_{\Omega_{mr}} \end{bmatrix} \begin{Bmatrix} b \\ p \\ a \\ q \\ \phi \\ \theta \\ r \\ w \\ v \\ u \\ \Omega_{mr} \end{Bmatrix} + \begin{bmatrix} \frac{B_{lat}}{tf} & \frac{B_{lon}}{tf} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \frac{A_{lat}}{tf} & \frac{A_{lon}}{tf} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & N_{\mu_y} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & T_{\mu_t} & 0 \end{bmatrix} \begin{Bmatrix} \mu'_{lat} \\ \mu'_{lon} \\ \mu'_t \\ \mu'_y \end{Bmatrix} \tag{4.16}$$

4.5 System Identification using CIFER®

As an initial step in the identification process, computing the *coherence* function can provide a measure of the extent to which an output is linearly related to some input over some frequency range. The magnitude squared coherence, $\gamma^2_{xy}(\omega)$ between two signals is given as:

$$\gamma^2_{xy}(\omega) \triangleq \frac{|R_{xy}(\omega)|^2}{|R_{xx}(\omega)|\,|R_{yy}(\omega)|} \tag{4.17}$$

where R_{xy} is the cross spectral density between the input and output, R_{xx} is the

auto-spectral density of the input and R_{yy} is the auto-spectral density of the output. An input/output pair with low coherence implies that either the input has no effect on the output or that the effect is entirely nonlinear. Conversely, an input/output pair with a high coherence can most likely be well modeled using a linear model such as a transfer function or state space model. Tischler[43] suggests a coherence above 0.6 for some useful frequency range is required for accurate transfer function identification. The DERIVID subroutine included in CIFER® utilizes coherence weighting to ensure frequency ranges with the best coherence contribute the most toward the transfer function estimate.

Given a flight data set with sufficient coherence, the software package CIFER® implements a series of subroutines that support frequency responses-based system identification. The first procedure, following data acquisition and scaling, was to apply the Frequency Response Identification (FRESPID)subroutine in CIFER® using multiple windows of appropriate lengths, which provided transfer function and coherence estimates. Considering recommendations in prior work[43], window lengths of 3, 6, 9, 12, and 15 seconds were used. A smaller window supports better averaging, however the largest window determines the lowest frequency that can be computed. The rotor/fuselage dynamics were assumed to be coupled, hence the lateral and longitudinal inputs influence the off-axis responses. To account for control cross-coupling, the (MISOSA) Multi-Input Conditioning subroutine was run to condition the data set considering the presence of multiple inputs. Following these procedures, the Window Combination subroutine (COMPOSITE) was run to appropriately combine the multiple responses. The resulting responses were the

processed and conditioned transfer function and coherence estimates used for state space model determination.

4.6 Results

The resulting estimated model compares well with the calculated transfer function estimates, suggesting that the relevant dynamics were captured. The transfer functions of the identified model are plotted with the computed transfer function estimates from the flight data in figures 4.7 through 4.19. The coherence functions plotted for each input/output pair indicate the degree to which the output is linearly related to the input at a particular frequency.

The identified state space model parameters, along with the associated error bounds, are given in Table 4.1. Percentage error Cramer-Rao (CR) bounds, as computed using CIFER's subroutine DERIVID, are included along with the computed insensitivity ($\bar{I}$) percentages. The Cramer-Rao bounds are theoretical minimum limits for the expected standard deviation σ in the parameters estimates that would be obtained from many experiments[43]. Tischler suggests valid parameter estimates are most likely to be obtained for the following conditions:

$$CR\% <= 20\% \qquad \bar{I}\% <= 10\% \tag{4.18}$$

The X_u and Y_v parameters exhibited the highest uncertainty, with X_u slightly exceeding the recommendations in 4.18. The identification of the Yamaha R-max helicopter discussed in Mettler's book also resulted in relatively high errors for the lat-

Table 4.1: Honeybee Identified Model Parameters with Cramer-Rao Error Estimate

parameter	value	CR %	$\bar{I}$ %
τ_f	0.15	5.467	1.996
B_a	1.55	5.829	1.712
L_b	1273	3.354	1.329
A_b	-2.82	10.94	3.569
M_a	341.6	2.672	0.9205
L_v	-8.246	10.94	3.569
M_u	3.599	8.497	3.217
Y_v	-0.4799	16.44	7.685
X_u	-0.5214	23.92	11.18
B_{lat}	0.245	2.408	1.114
B_{lon}	0.043	9.280	4.277
A_{lat}	0.044	7.309	2.974
A_{lon}	-0.202	2.455	1.053
N_r	-0.8786	5.342	2.472
Z_w	-0.6802	6.792	2.784
$Z_{\Omega_{mr}}$	0.170	6.447	2.484
$T_{\Omega_{mr}}$	-6.182	4.362	1.142
N_{μ_y}	39.06	4.407	2.189
T_{μ_t}	1449	5.081	1.576
Δt_{cyclic}	0.033	7.884	3.550
Δt_{μ_t}	0.1163	1.686	0.602
Δt_{μ_y}	0.069	2.551	1.187
g	9.81	-	-

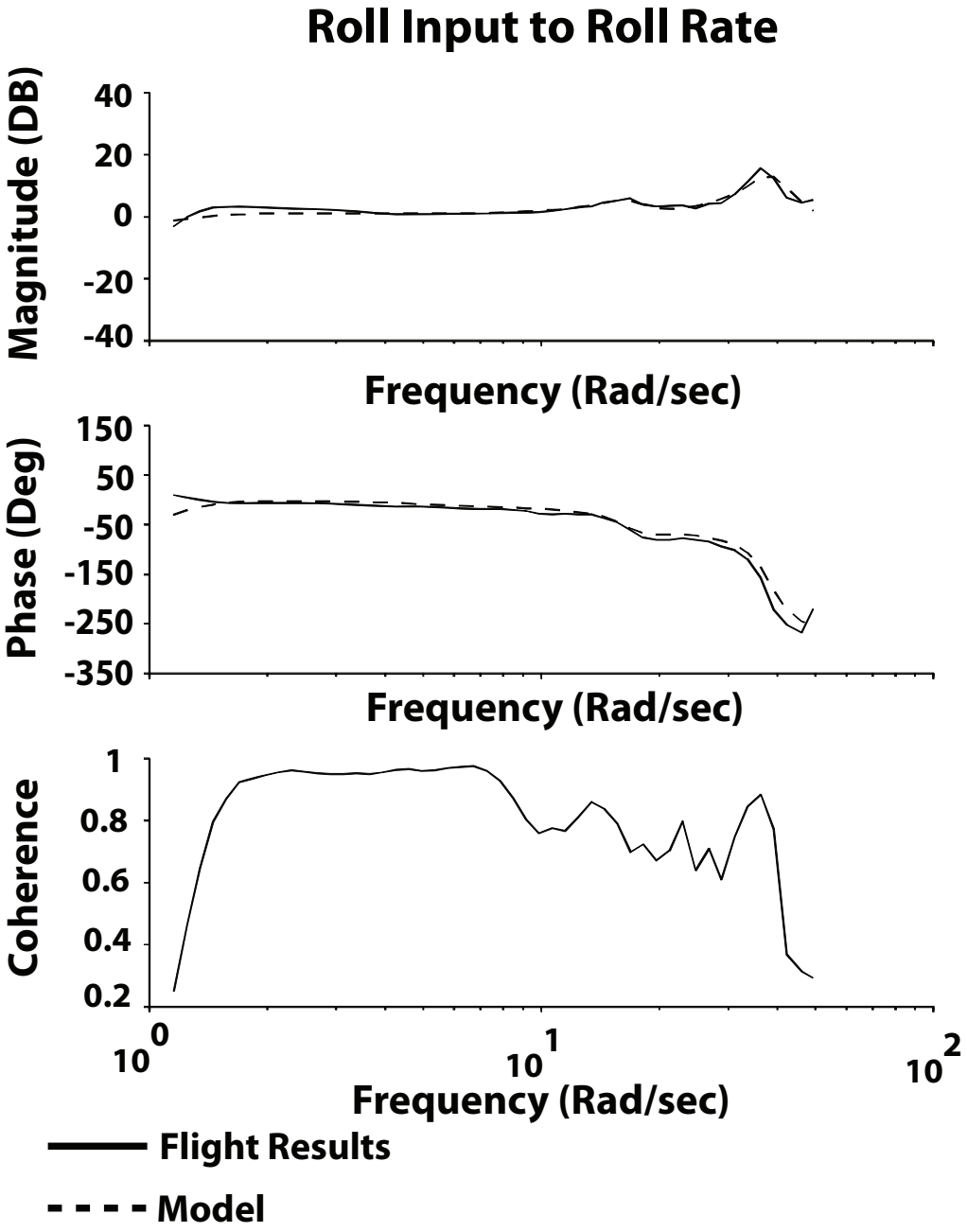

Figure 4.7: Roll Input to Roll Rate Transfer Function.

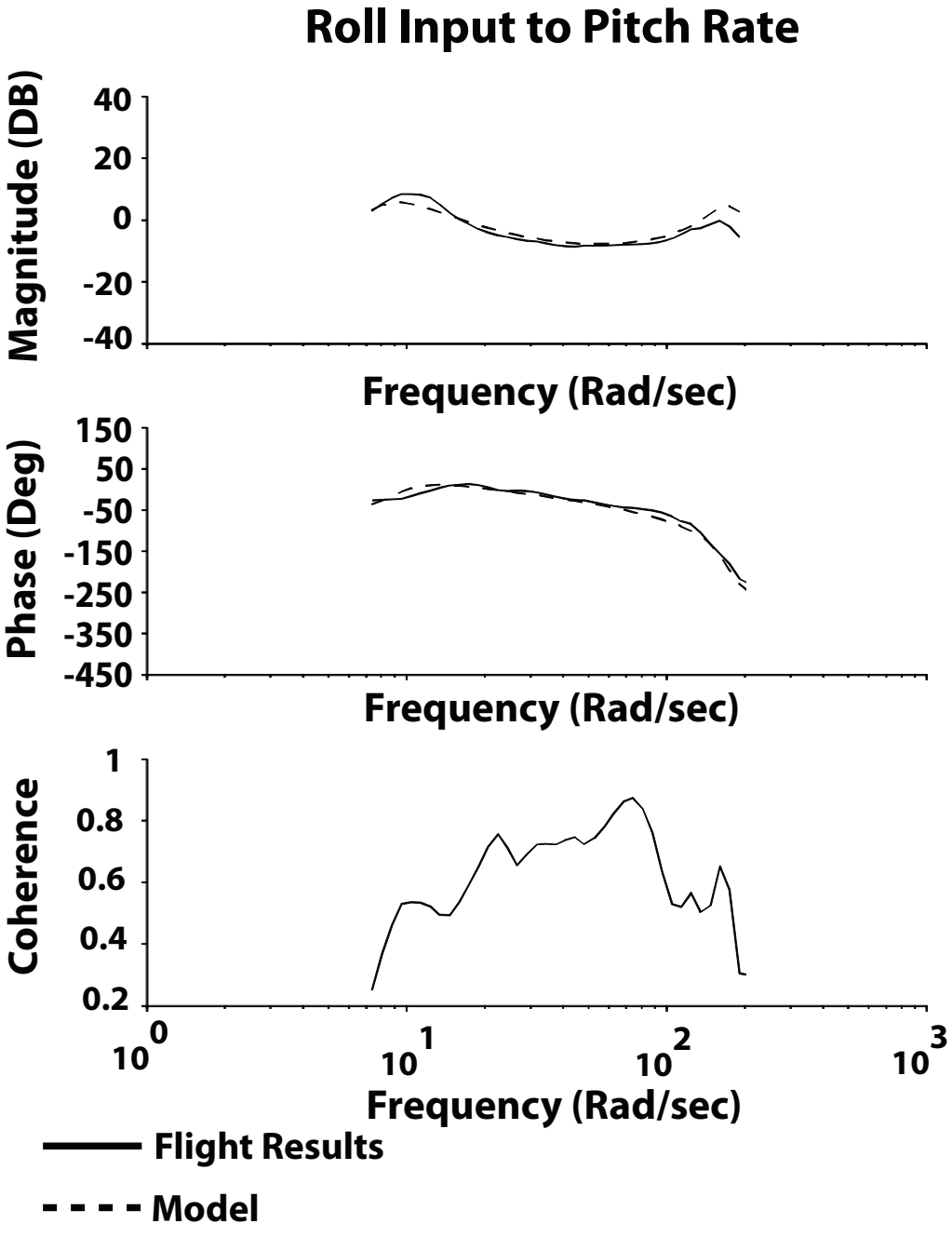

Figure 4.8: Roll Input to Pitch Rate Transfer Function.

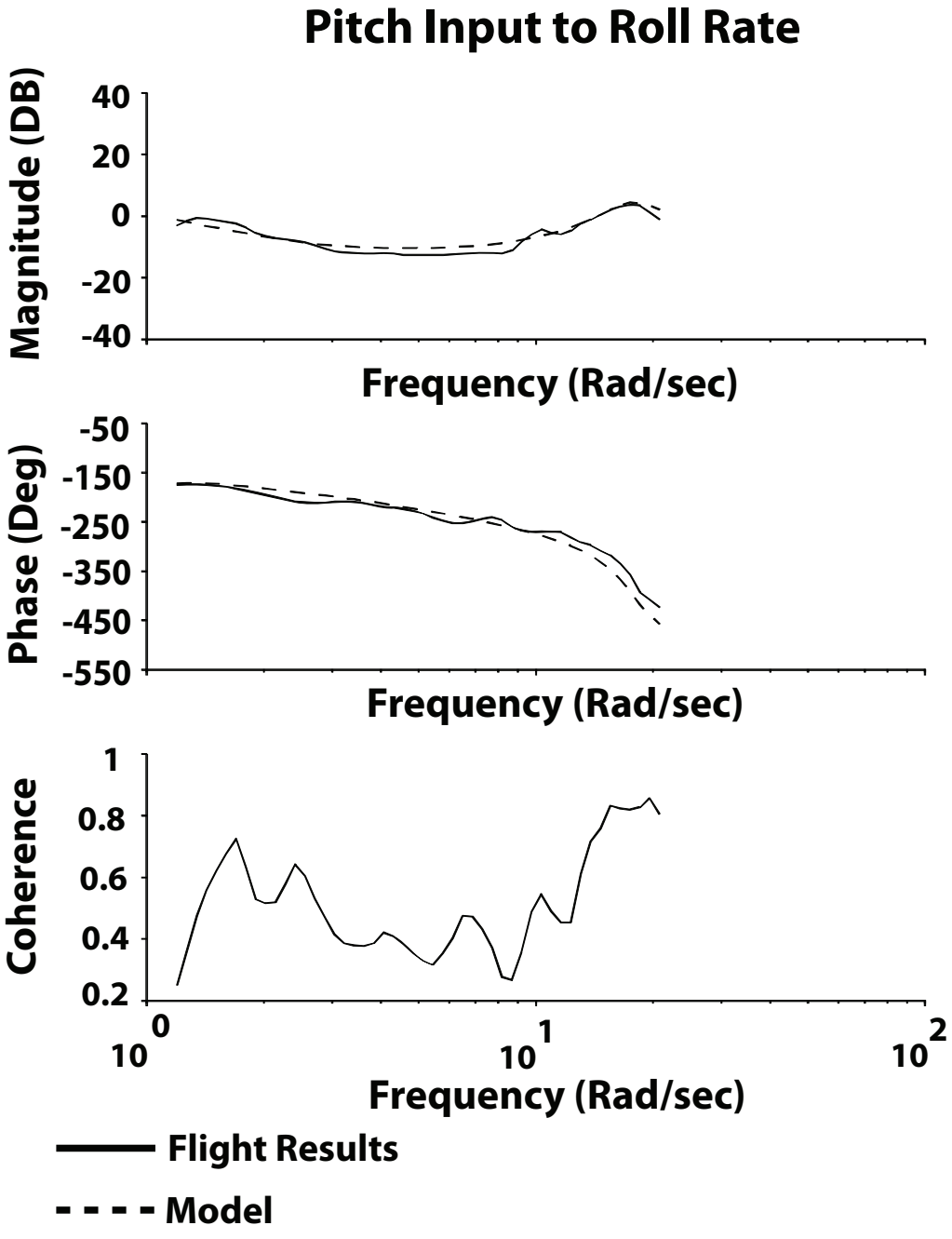

Figure 4.9: Pitch Input to Roll Rate Transfer Function.

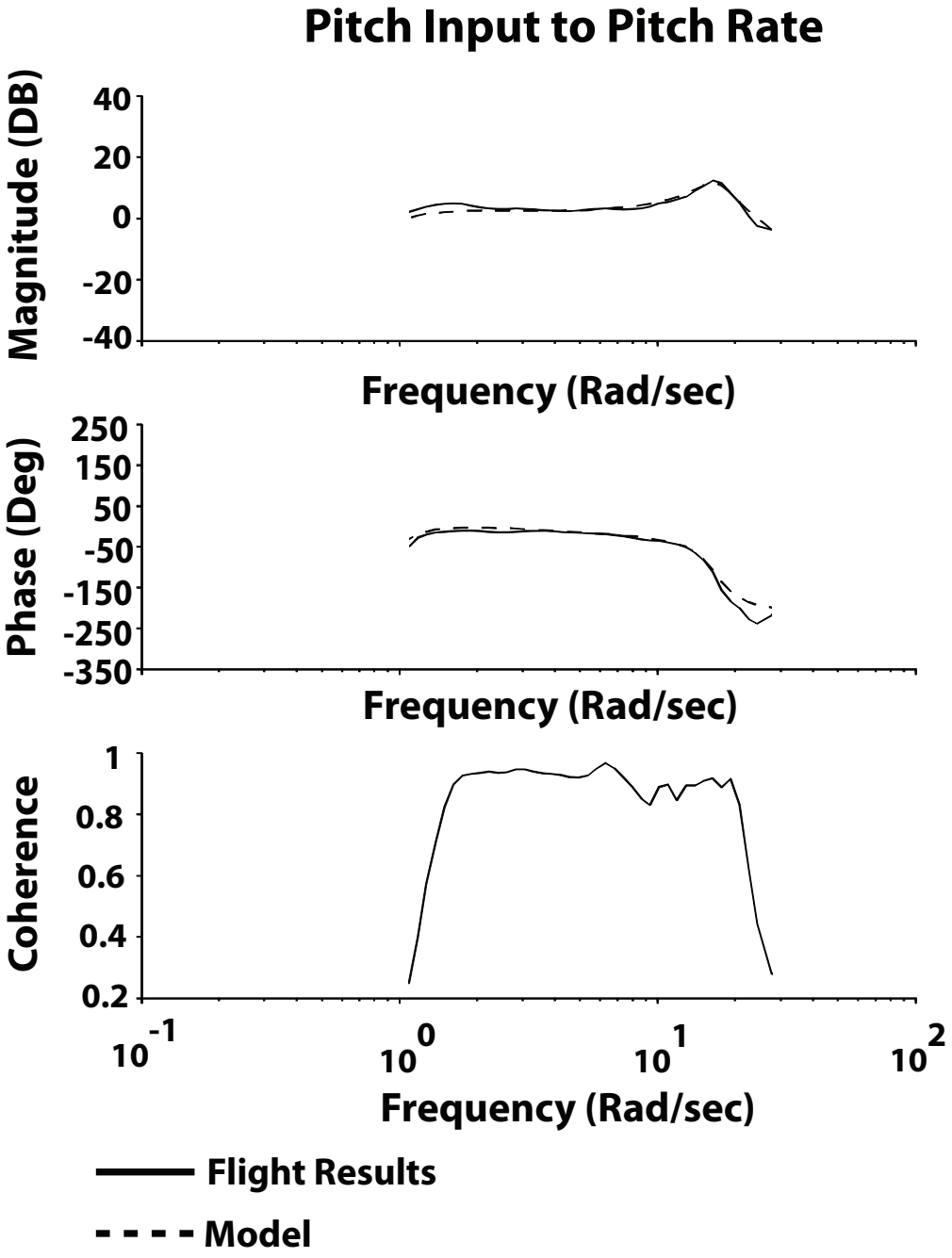

Figure 4.10: Pitch Input to Pitch Rate.

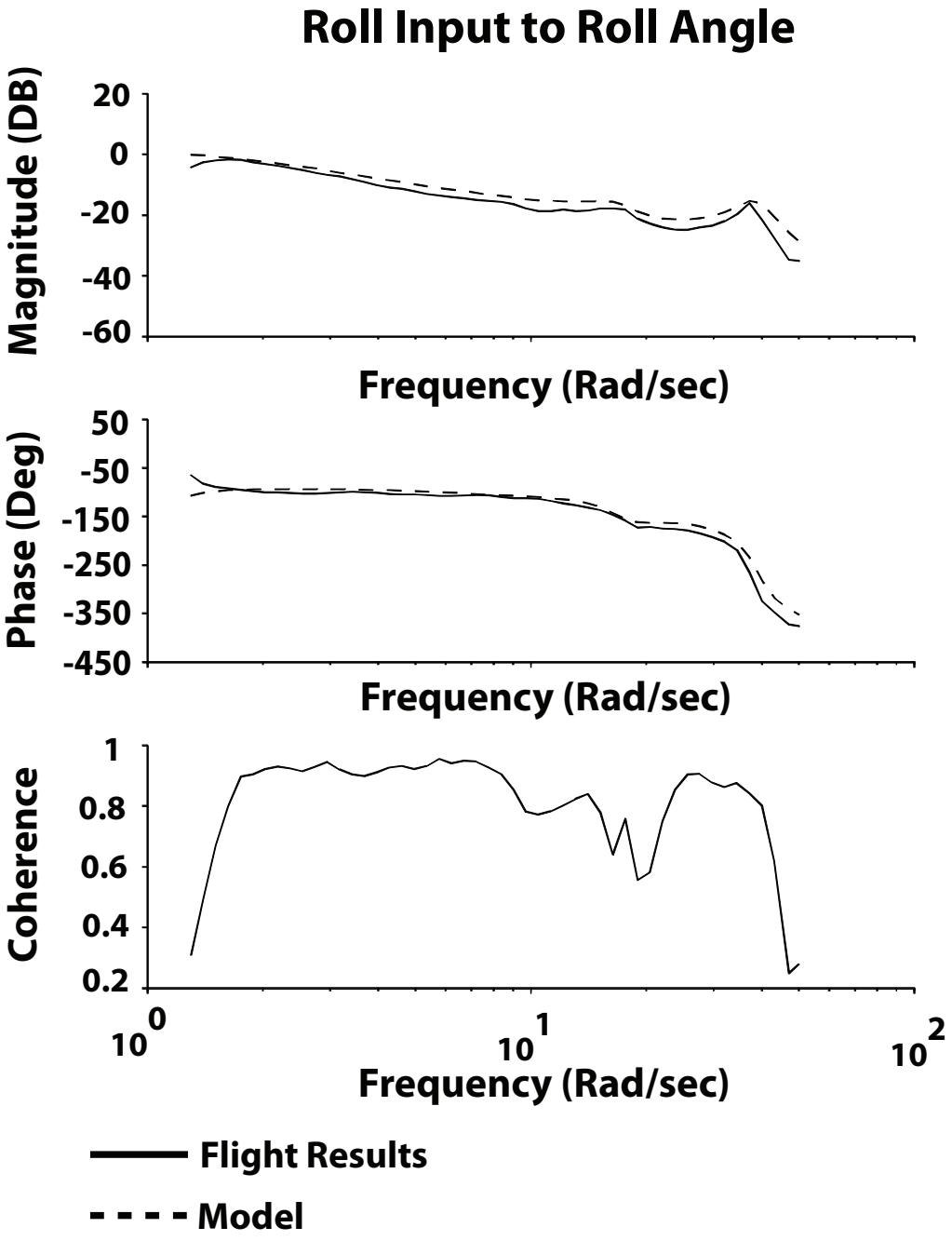

Figure 4.11: Roll Input to Roll Angle Transfer Function.

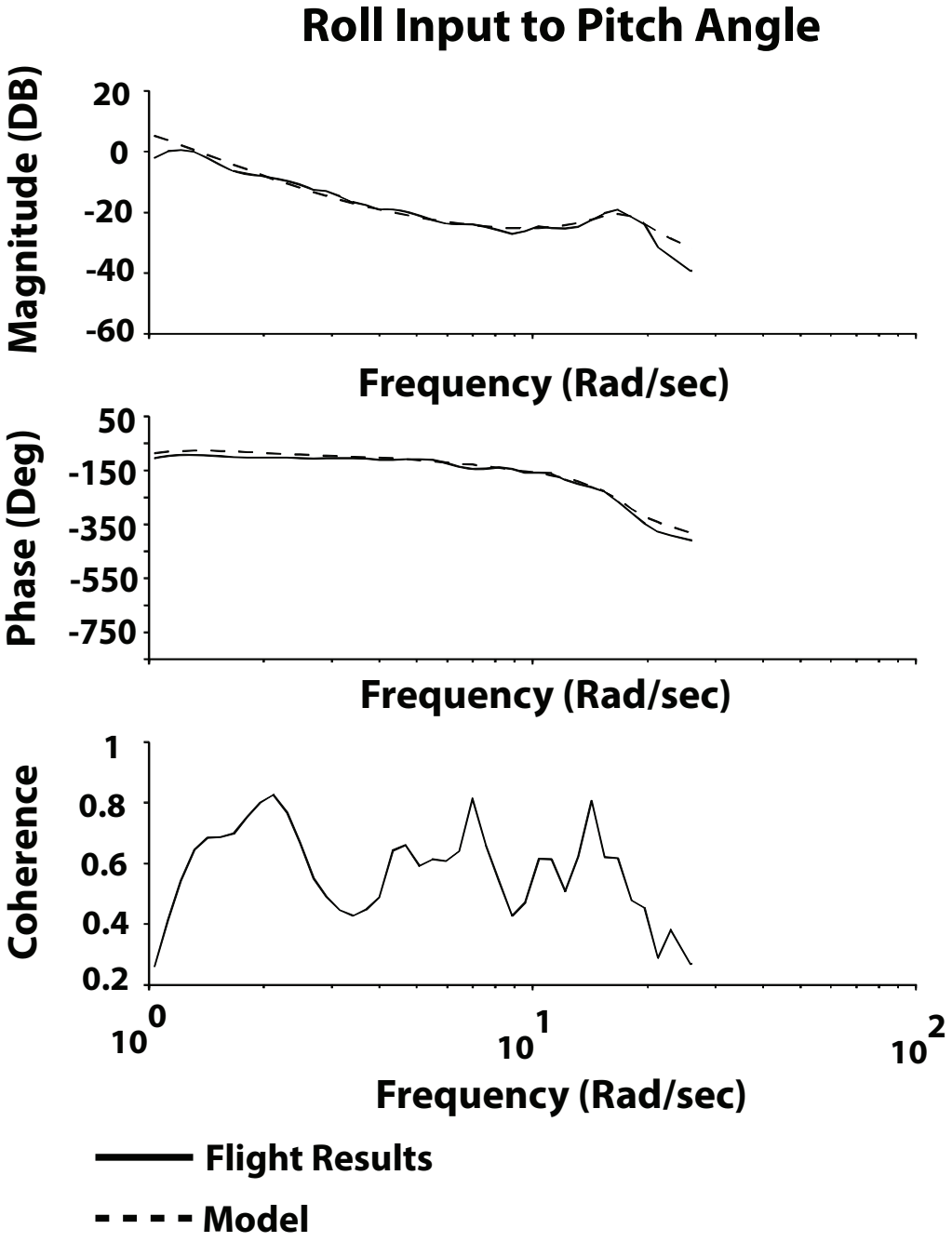

Figure 4.12: Roll Input to Pitch Angle Transfer Function.

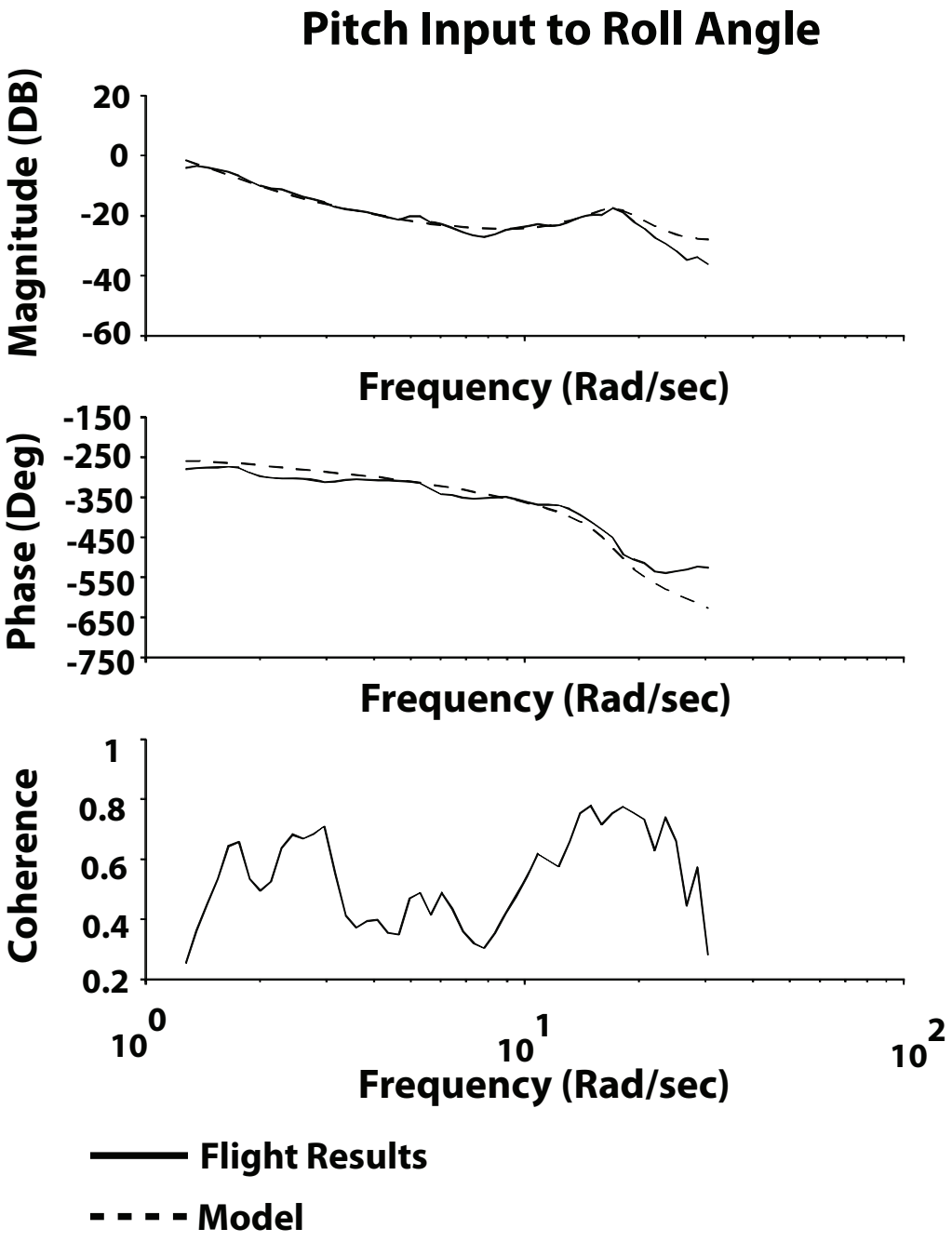

Figure 4.13: Pitch Input to Roll Angle Transfer Function.

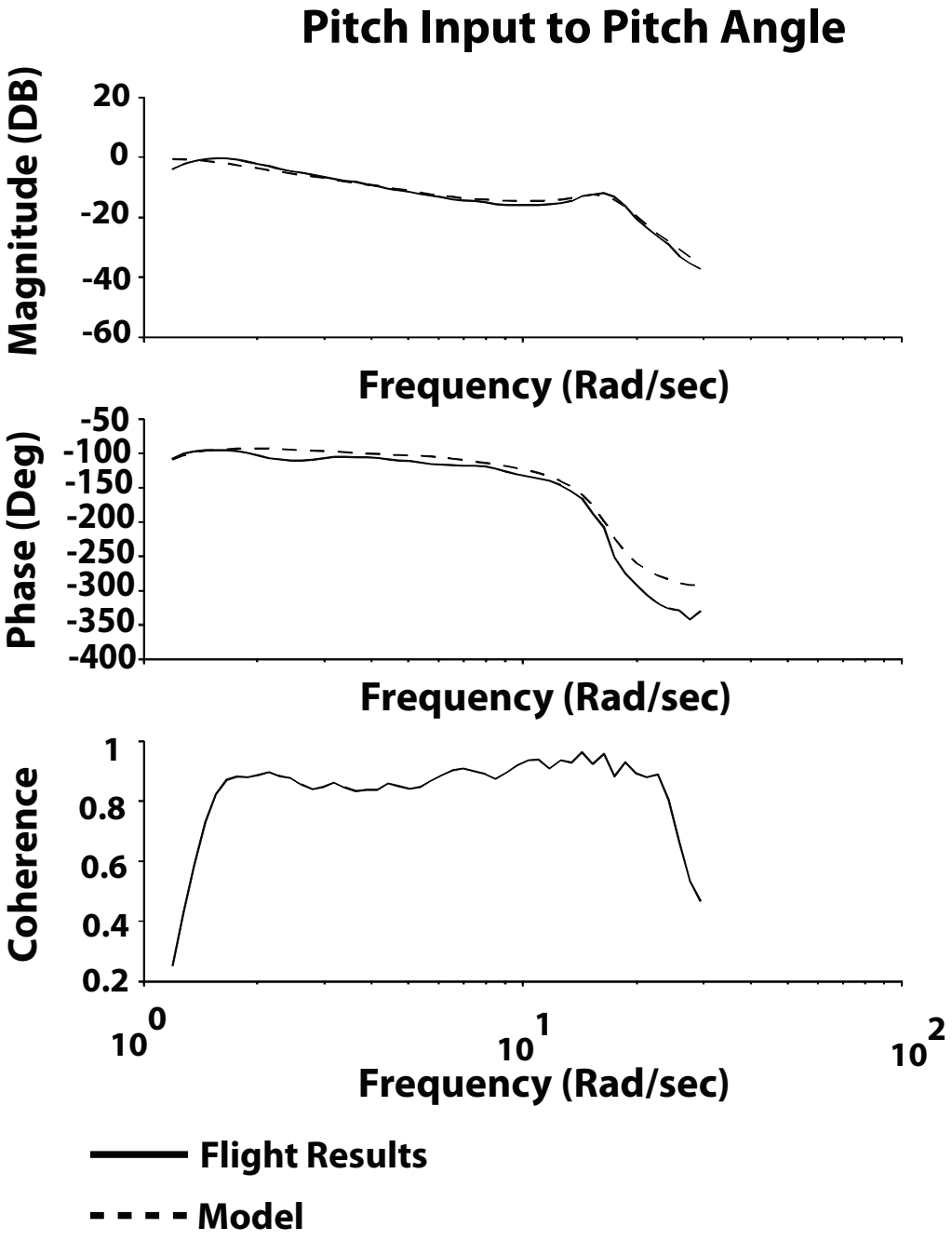

Figure 4.14: Pitch Input to Pitch Angle Transfer Function.

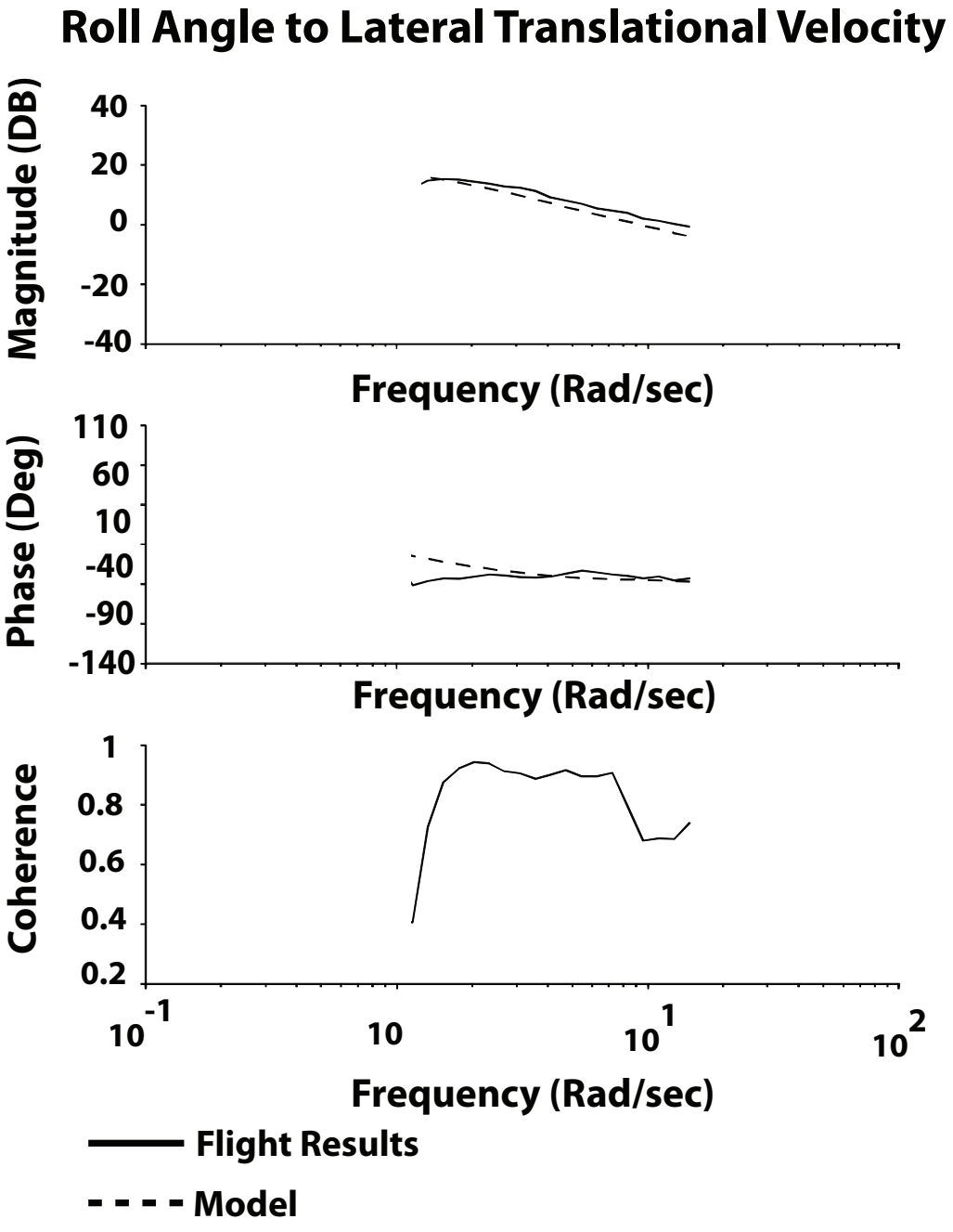

Figure 4.15: Roll Angle to Lateral Velocity Transfer Function.

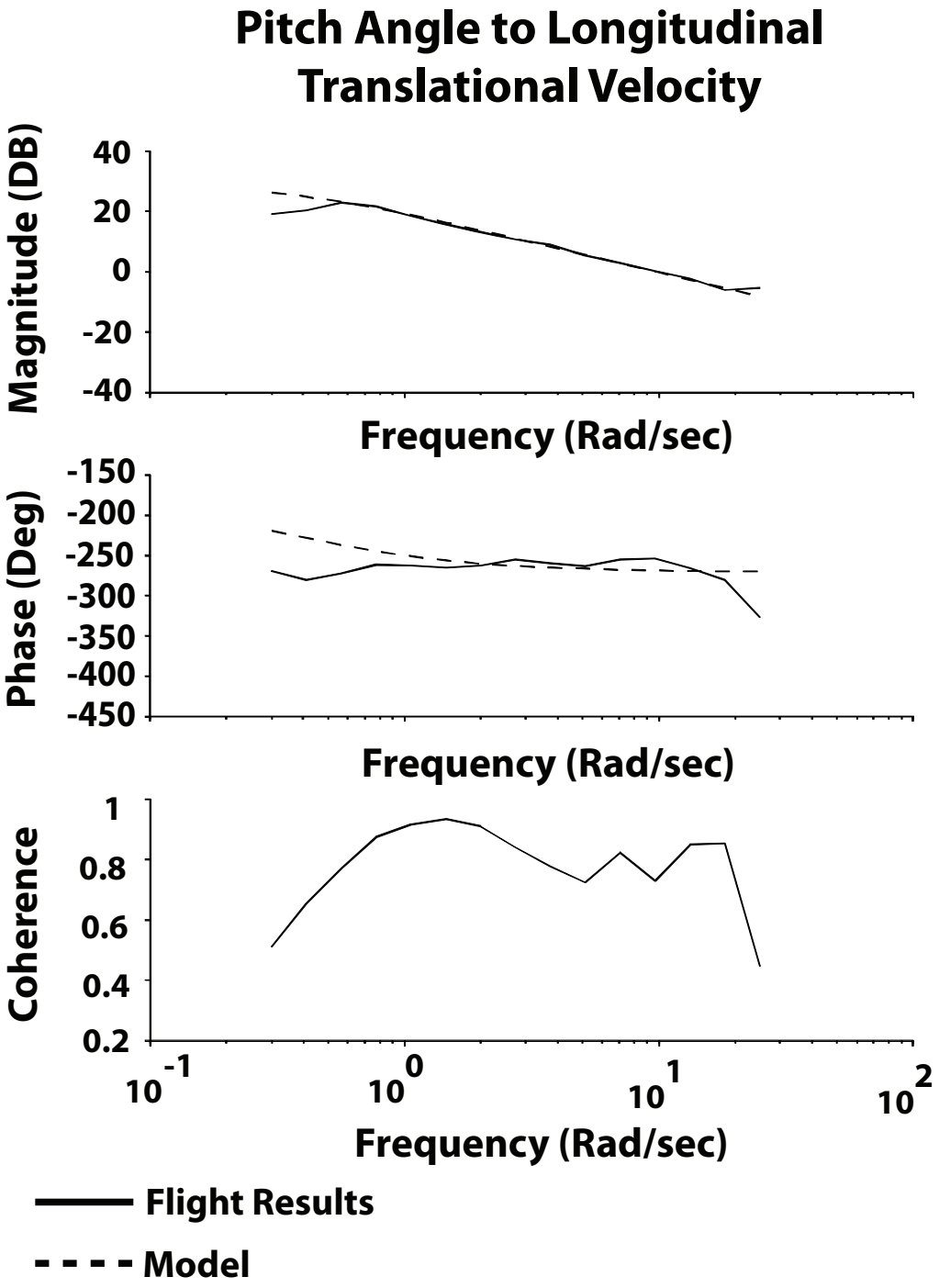

Figure 4.16: Pitch Angle to Longitudinal Velocity Transfer Function.

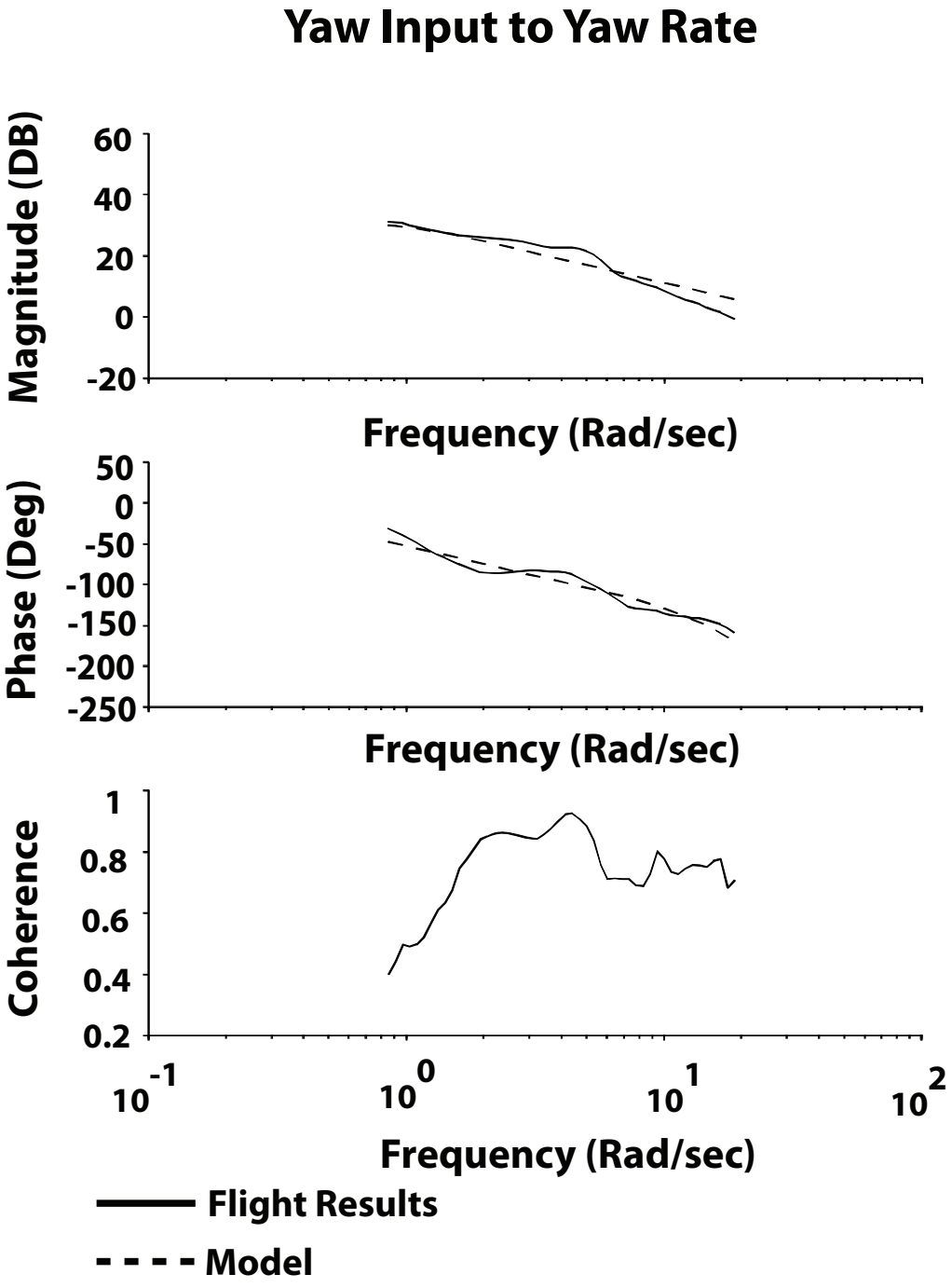

Figure 4.17: Yaw Input to Yaw Rate Transfer Function.

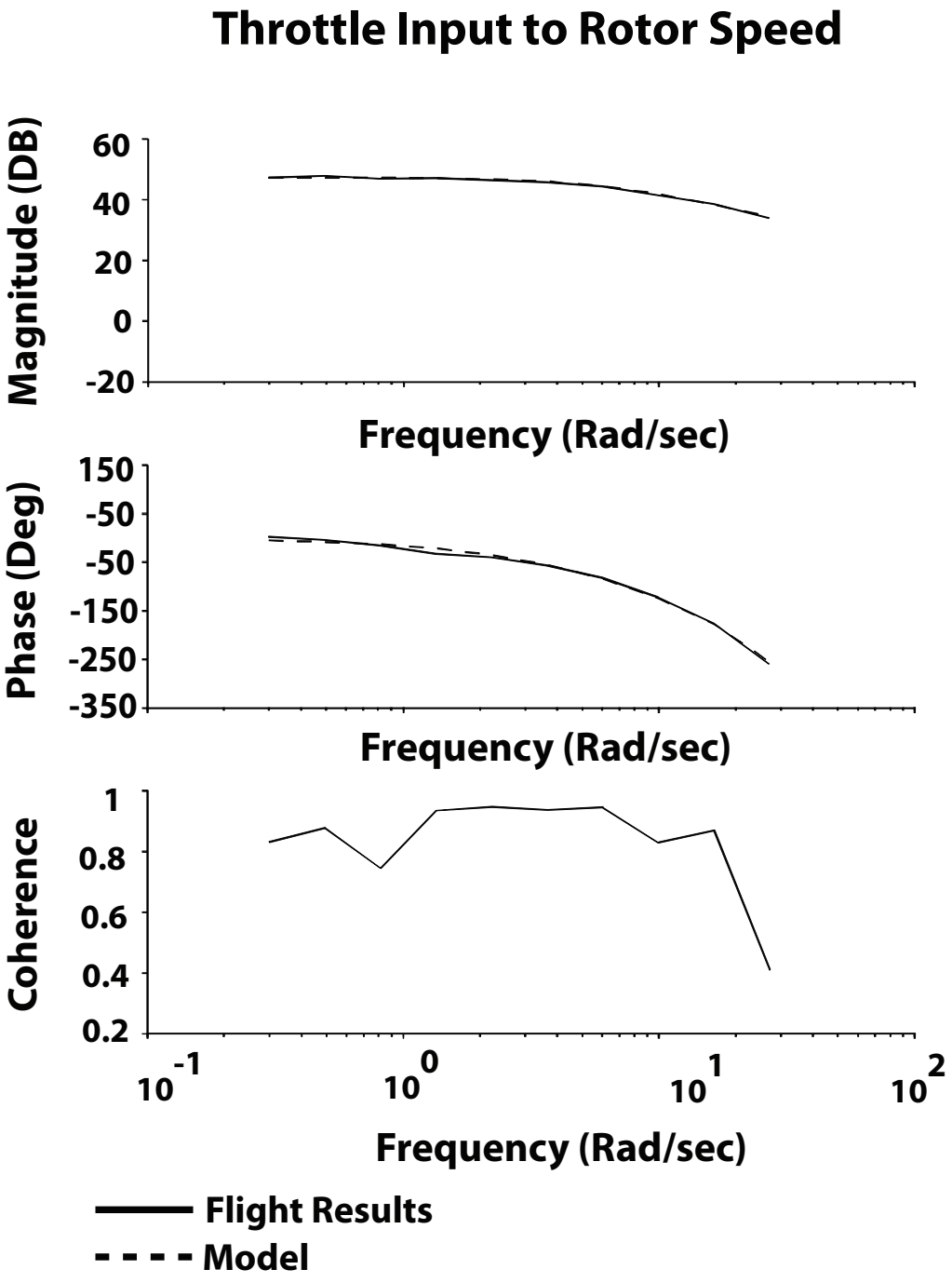

Figure 4.18: Throttle Input to Rotor Speed Transfer Function.

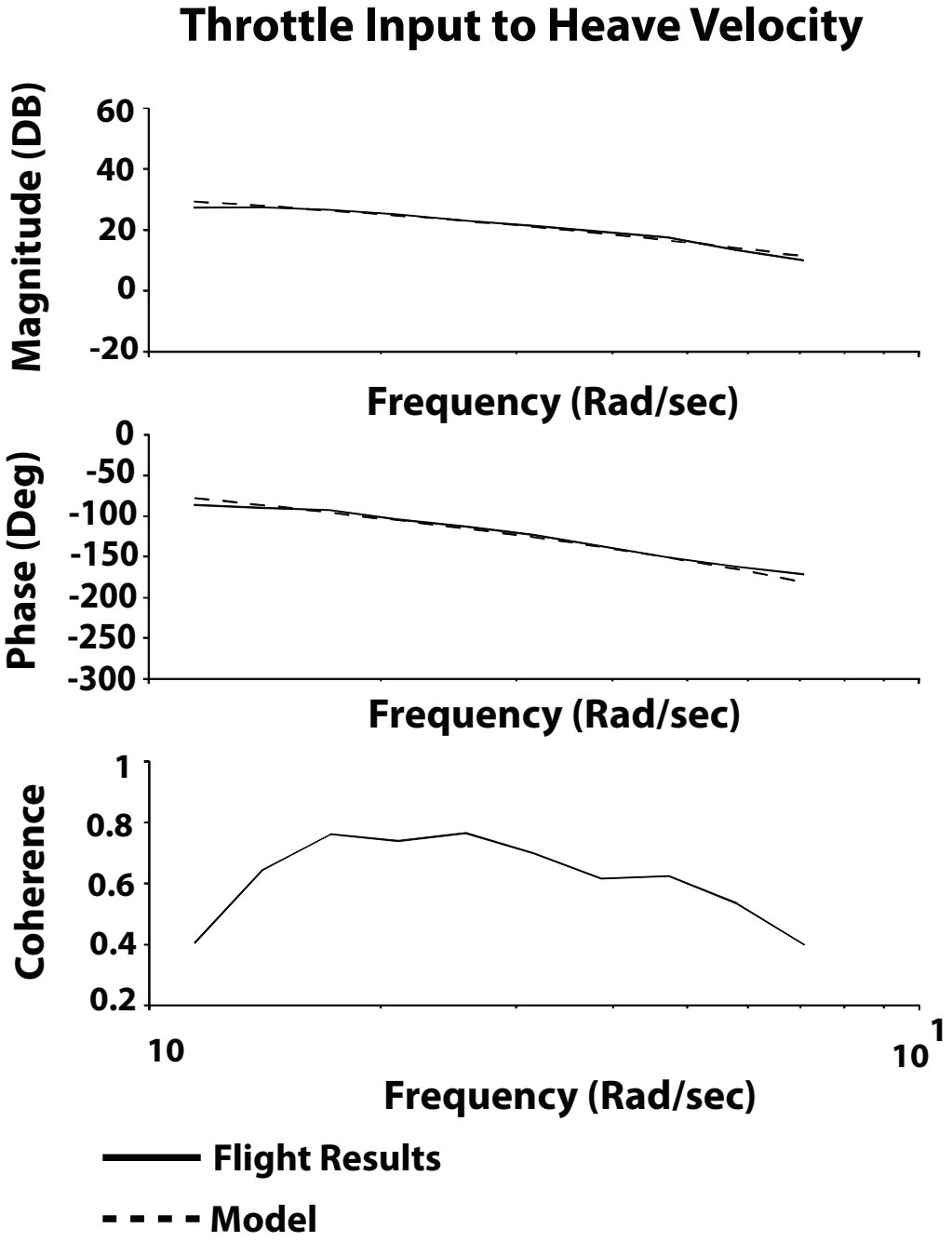

Figure 4.19: Throttle Input to Heave Velocity Transfer Function.

eral and longitudinal stability derivatives, Y_v and X_u, suggesting poor low-frequency content in the flight data as the cause. This assumption seems consistent with this research as well, as flight tests were carried out in an indoor facility without the physical space required to obtain very low frequency translational velocity flight data.

Note that both the resonant frequencies of the rotor/fuselage dynamics can be observed in the roll (lateral) input to roll rate and roll angle, however the pitch input to pitch rate and pitch angle computed transfer functions did not exhibit sufficient coherence to observe both resonant frequencies. The relatively low roll inertia of the vehicle resulted in a relatively high roll bandwidth of approximately 35 rad/s as compared to the pitch dynamics where the bandwidth is closer to 17 rad/s.

Additionally, the low frequency phugoid dynamics cause a slight increase in the magnitude of the rate responses observed at the lower frequencies. This effect is most clearly observed in the off-axis rate responses creating a noticeable "bucket-like" magnitude response.

4.7 Time Domain Verification

One qualitative method of checking how well a model predicts the dynamics is to simulate the model using the input signals acquired from a flight test, and then compare the predicted output with the actual output. Time domain comparisons were performed using the VERIFY subroutine included in the CIFER package. This subroutine estimates the input and output bias values to minimize the error accu-

mulated by integrating the model numerically. Figures 4.20 through 4.23 present representative time domain comparisons for each of the available outputs. These plots demonstrate that the model does capture the major dynamic behavior of the vehicle in the frequency range of interest.

4.8 Analysis of Vehicle Dynamics

The eigenstructure of the linearized model can provide much information about the behavior of the vehicle, as observation of the eigenvector makeup allows an examination of which state variables are dominant in each mode. The eigenvectors were computed according to the traditional left eigenvector equation in equation 4.19, where A is the system matrix, λ is the vector of eigenvalues, and I is the identity matrix.

$$\det[\lambda \mathrm{I} - A] = 0 \tag{4.19}$$

The lateral and longitudinal dynamics were developed and identified separately from the heave and yaw dynamics, as low coherences suggested very little dynamic interaction between these subsystems. These three subsystems can therefore be analyzed separately. The model eigenvalues, with the associated mode names, are given in table 4.2. The yaw rate mode is a simple first order response. The heave dynamics, including the heave velocity w and rotor speed Ω_{mr}, constitute a second order over-damped response where both poles are on the negative real axis. The eigenvalues for the lateral and longitudinal dynamics, however, are fully coupled and

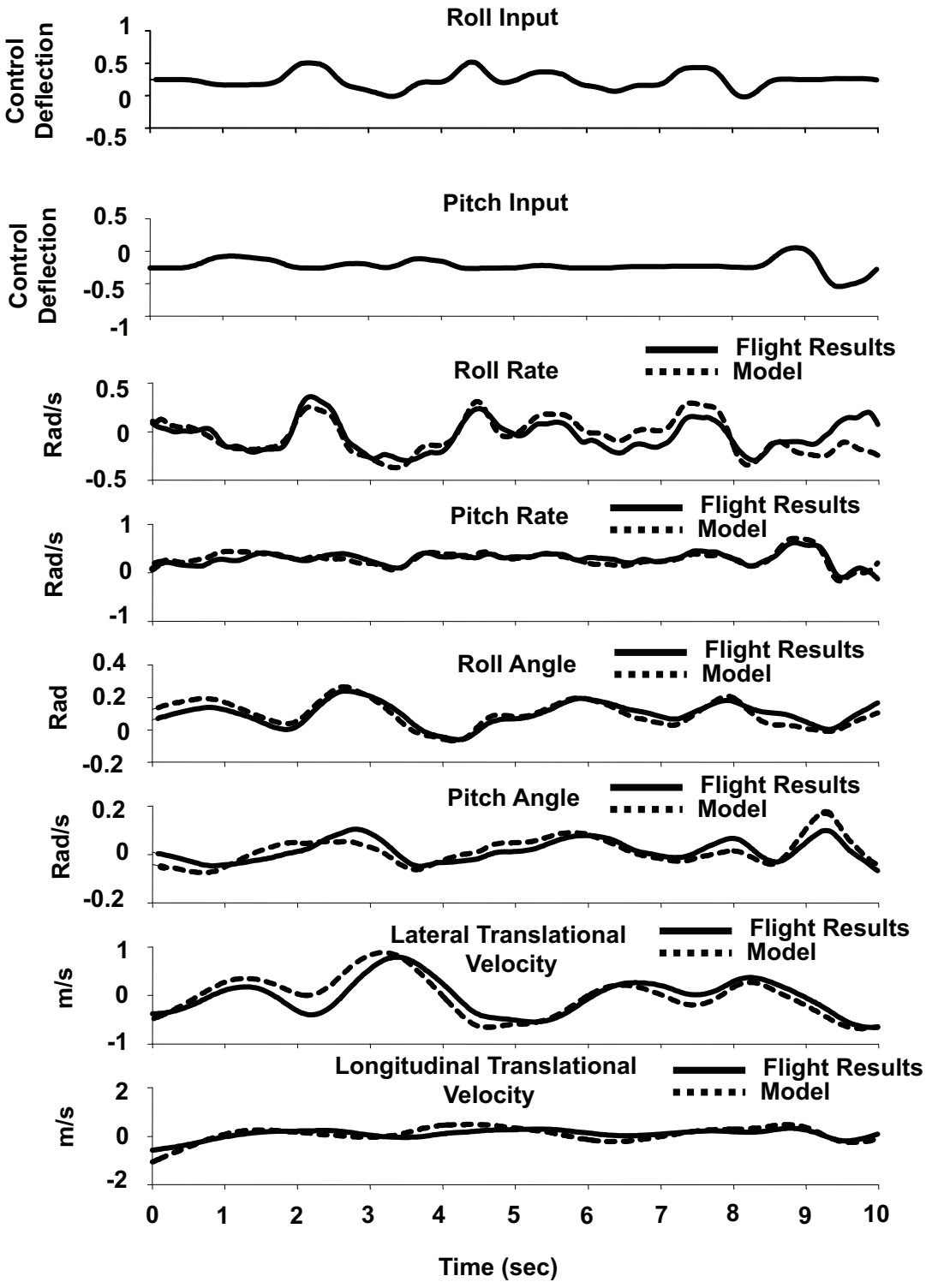

Figure 4.20: Time Domain Comparison with Lateral Excitation.

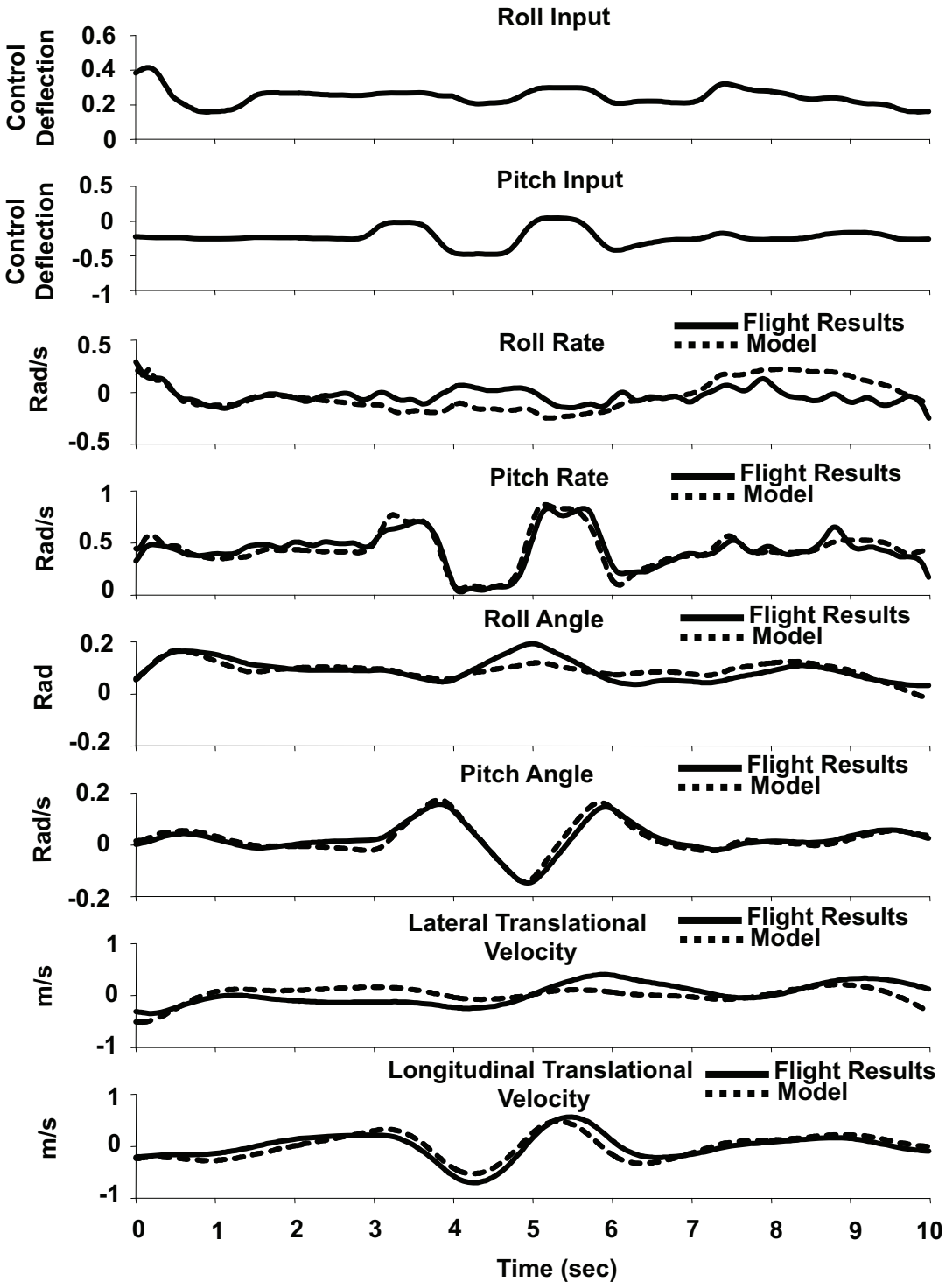

Figure 4.21: Time Domain Comparison with Longitudinal Excitation.

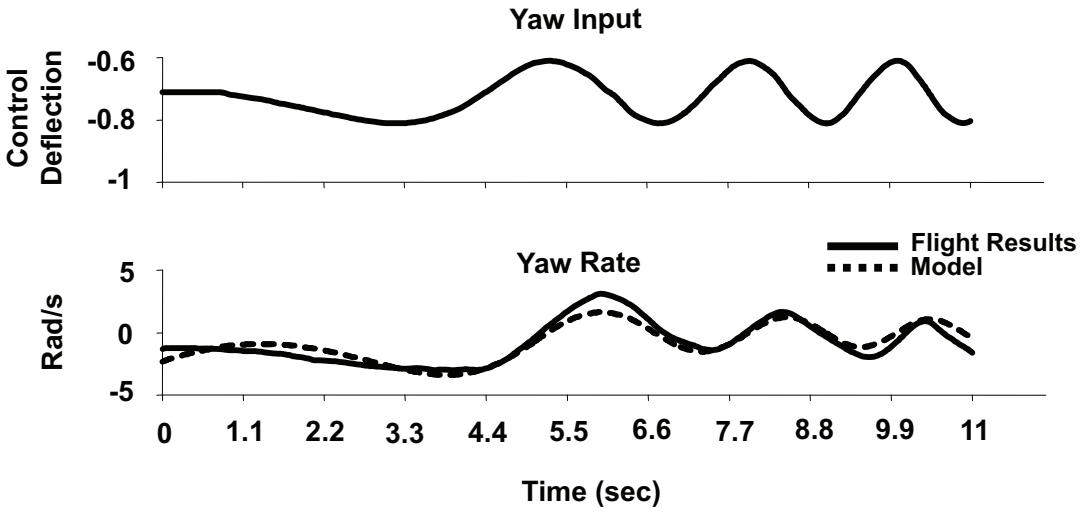

Figure 4.22: Time Domain Comparison with Yaw Excitation.

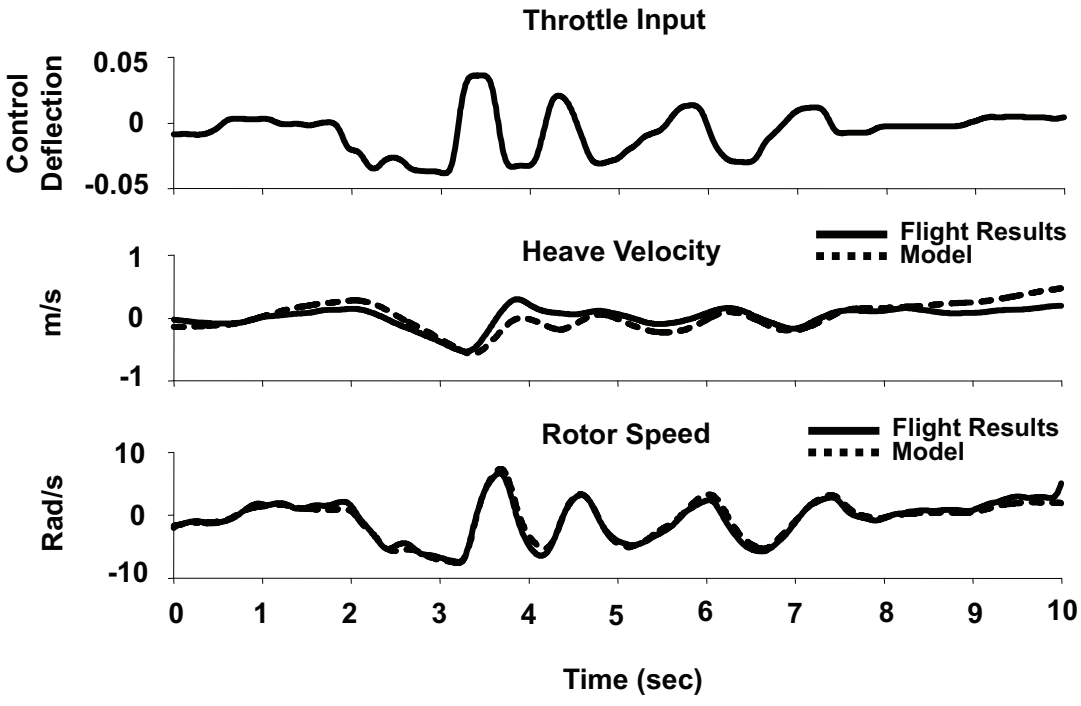

Figure 4.23: Time Domain Comparison with Heave Excitation.

Table 4.2: Eigenvalues of Identified Model

eigenvalue no.	value (Rad/s)	mode
1	-3.71+38.57i	lateral flapping
2	-3.71-38.57i	lateral flapping
3	-2.71+16.76i	longitudinal flapping
4	-2.71-16.76i	longitudinal flapping
5	-0.87+0.86i	stable phugoid
6	-0.87-0.86i	stable phugoid
7	0.28+0.84i	unstable phugoid
8	0.28-0.84i	unstable phugoid
9	-0.879	yaw mode
10	-0.68	heave velocity mode
11	-6.161	rotor speed mode

Table 4.3: Eigenvectors for Rotor/Fuselage Dynamics of Identified Model

-0.003+0.03i	-0.003-0.03i	0.008+0.003i	0.008-0.003i
1	1	0.091-0.63i	0.091+0.63i
-0.018-0.003i	-0.018+0.003i	0.008-0.05i	0.008+0.05i
-0.01+0.16i	-0.01-0.16i	-1	-1

are worth further examination.

The eigenvectors associated with only the rotor/fuselage dynamics are given in table 4.3. Note each column was scaled such that the maximum absolute value would be one to facilitate comparisons. Eigenvalues 1-4 in table 4.2 are associated with the eigenvectors in table 4.3. The pure lateral flapping mode does not induce as much pitch rate and pitch flapping as a pure longitudinal flapping induces a roll rate and lateral flapping. Since the rotor characteristics are the same for the lateral and longitudinal degrees of freedom, this is a direct result of the lower lateral inertia, I_{xx}, as compared to the longitudinal inertia, I_{yy}. The roll rate of the lateral flapping model exhibits approximately 20% coupling with the pitch axis, and the pitch rate in the logitudinal mode exhibits close to 65% coupling with the roll rate.

Including the control sensitivity matrix however, reveals an additional source of coupling that originates from the actuators. The amount the pitch and roll inputs each affect the pure lateral and longitudinal flapping modes can be given as:

$$B_\xi = V^{-1}B \tag{4.20}$$

where V is the matrix of eigenvectors, B is the model's control sensitivity matrix, and B_ξ is the control sensitivity matrix transformed into modal coordinates, and is shown in table 4.21. Here, it can be seen that an input to the lateral control would result in an approximate 30% excitation of the longitudinal flapping mode. Conversely, an input to the longitudinal control would induce a 45% excitation in the lateral flapping mode. It is speculated that this high degree of coupling exhibited in the control matrix is mostly due to gyroscopic forces, as the B matrix captures coupling due to applied forces. The coupling exhibited by the A matrix eigenvectors is assumed to be aerodynamic in origin as it arises from induced rates.

$$B_\xi = \left[\begin{array}{c|c} -2.22 - 24.60i & 6.75 - 4.3i \\ -2.22 + 24.60i & 6.75 + 4.3i \\ 3.93 + 2.94i & 0.62 - 10.8i \\ 3.93 - 2.94i & 0.62 + 10.8i \end{array}\right] \tag{4.21}$$

A pole map is shown in figure 4.24 that shows how the eigenvalues, or poles, of the Honeybee model compare to the poles of the Yamaha R50, as identified by Mettler[39]. The rotor/fuselage poles of the Honeybee exhibit a slightly lower damping ratio than those of the Rmax, but act on a quicker time scale, as intuitively expected. The unstable eigenvalues of the Honeybee have a slightly lower real part than the R50. As this mode is due to M_u and L_v, a smaller rotor coning angle would explain this, as a lower coning angle would result in a lower moment induced by translational velocity. The lower damping ratio of the phugoid dynamics is a result of the lower identified values for the Y_v and X_u derivatives, resulting in lower

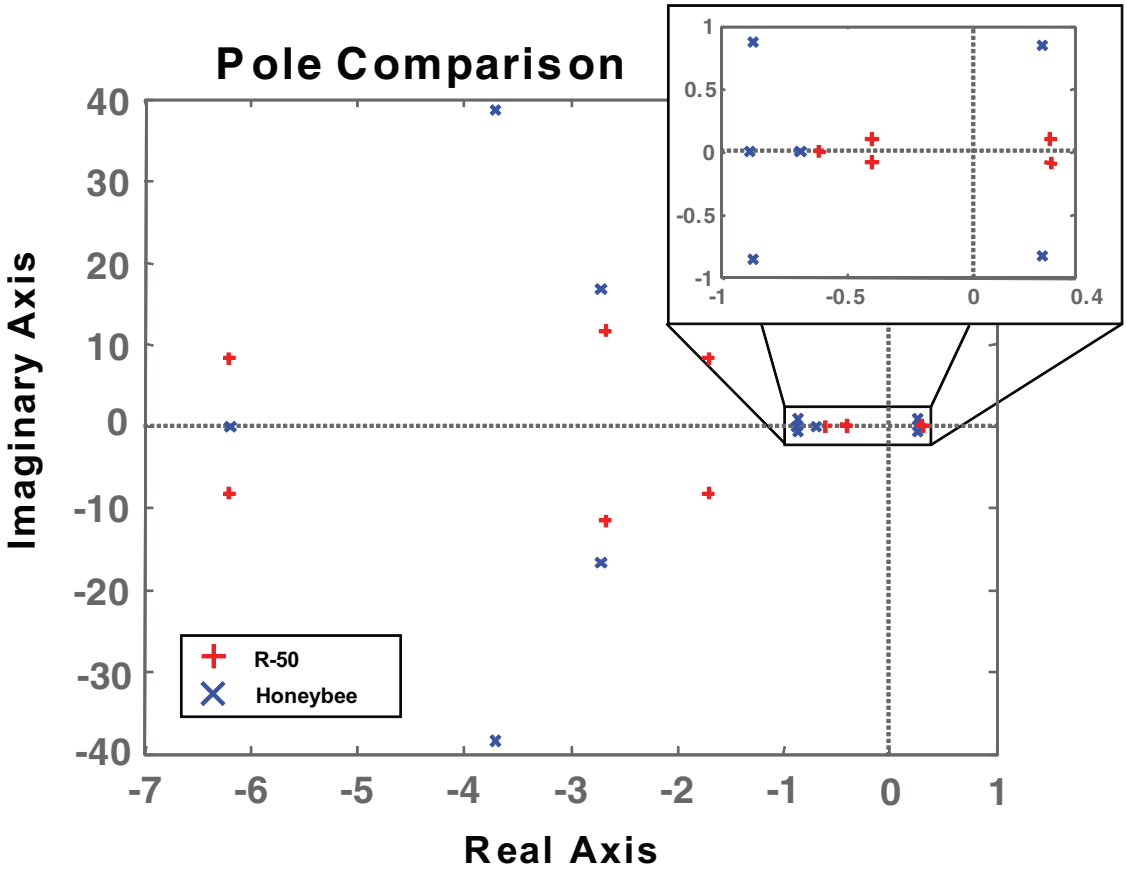

Figure 4.24: Pole Comparison between Yamaha R-max and Honeybee

Table 4.4: Froude Values for Various Vehicles

Vehicle	Froude No.
Honeybee	541
UH-1H	862
R22	1134
R-50	1230
X-Cell	2165

translational velocity damping. The lower translational velocity damping is not surprising as the Honeybee does not have a large fuselage profile, which is typically a large contributing factor.

Froude scaling was employed to compare the Honeybee to larger counterparts in a more meaningful way. The Froude number can be given as[39]:

$$F = \frac{(\Omega R)^2}{Rg} \tag{4.22}$$

where Ω is the rotor speed, R is the rotor radius, and g is gravitational acceleration.

A table of computed Froude numbers computed for various helicopters is given in table 4.4. If two vehicles are dynamically similar, determined by comparing their Froude numbers, then scaled comparisons between vehicle parameters can be considered physically meaningful. The Honeybee was compared to the UH-1H, as their Froude numbers are the most similar. The analysis proceeded by scaling the

Table 4.5: Froude Comparison between UH-1H and Honeybee

Parameter	Honeybee	Honeybee (scaled)	UH-1H
rotor radius (ft)	0.83	24	24
weight (lb)	0.86	20800	8000
rotor speed (rad/s)	145	27	34
L_b	1273	44.1	19.2
ω_n (rad/s)	35	6.51	4.38
$\tau_f\Omega$	21.75	21.75	5.7

Honeybee up by matching the rotor radii. Table 4.5 shows some original and scaled Honeybee parameters as compared to the UH-1H. The parameter $\tau_f\Omega$ is the non-dimensional rotor time constant, ω_n is the natural frequency of the roll rate response, and the stability derivative L_b is the lateral rotor flapping stiffness parameter.

Additionally, the scaled version of the Honeybee would weigh over two and a half times as much as the UH-1H. Small micro air vehicles typically require a higher fraction of the vehicle dedicated to the power systems and structure as compared to scaled up counterparts, therefore it is not unexpected that the empty weight fraction of the Honeybee is high. Other scaled terms confirm expected results. The L_b term is over twice the corresponding value for the UH-1H, indicating a high comparative stiffness. Likewise, the natural frequency of the scaled Honeybee rotor/fuselage dynamics is 1.5 times higher and the non-dimensional rotor flap constant is 3 times that of the UH-1H. The scaled rotor speed, however is 80% that of the UH-1H.

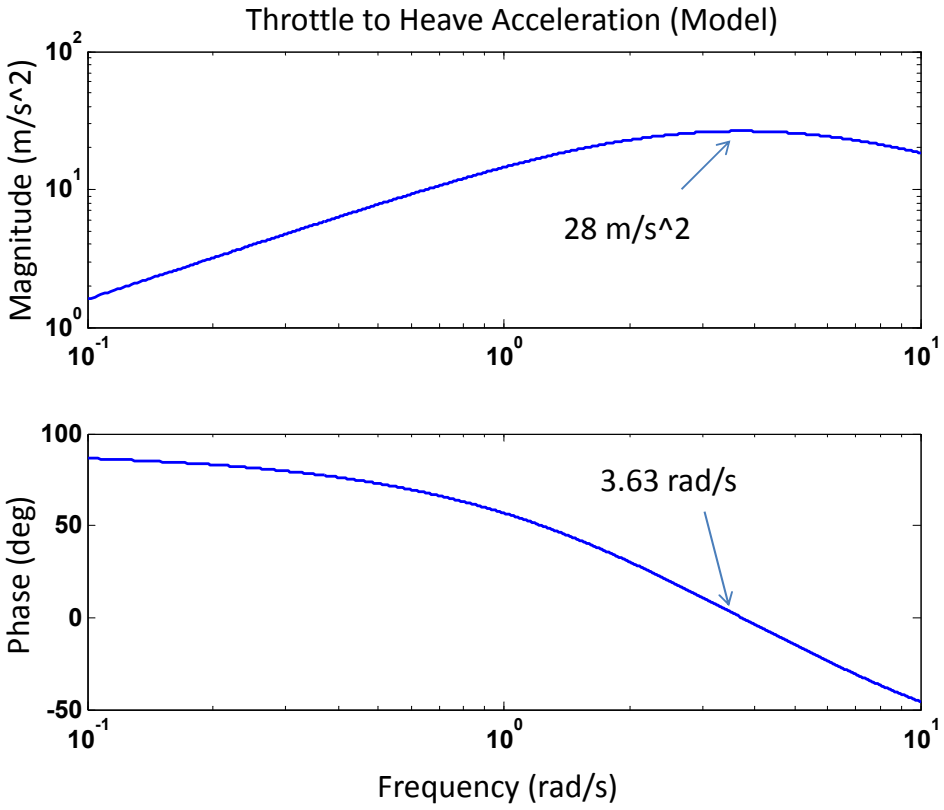

Figure 4.25: Transfer Function from Throttle to Heave Acceleration

Examining the heave dynamics can also provide information regarding the vehicle's maneuverability. The transfer function from throttle to heave velocity can be used to estimate the maximum acceleration possible at the linearized flight condition. The derivative of the heave velocity is the acceleration, the maximum value of which is the thrust divided by the vehicle's weight. The plot of the model $s*TF_{\mu t->w}$, seen in Figure 4.25, reaches a maximum of 28 m/s^2 (2.8g) at approximately 3.6 rad/s. This estimate can be compared to the identified heave accelerations for the X-Cell and R-50 helicopters of 21 m/s^2 and 11 m/s^2, respectively.

While it is intuitively expected that a small vehicle would exhibit faster dy-

namics than a larger vehicle, this analysis shows that the vehicle is more maneuverable even when accounting for size. This is due to the fast control responses, high thrust to weight, and hingeless rotor design. The dynamics are, however, quite representative of a rotary wing vehicle, hence the same *hybrid model* utilized for larger counterparts can be successfully employed on vehicles at this scale.

4.9 System Identification Summary

Using frequency response identification techniques, a state space model of a small electric helicopter, linearized about hover, was identified. The identification showed that the *hybrid model* structure, previously successfully utilized to model the rotor/fuselage interaction on larger vehicles, has been shown to also successfully work for small micro rotorcraft. The dynamics of the vehicle are similar in structure to that of a larger helicopter, however evolve on a much faster time scale. Finally, this represents one of the first vehicle identifications performed using a visual tracking system for kinematic measurements.

Chapter 5

Feedback Control

Typically, attitude control for rotorcraft is implemented by first controlling the attitude dynamics, then placing that control loop within an outer translational loop, resulting in a cascaded control loop architecture. In chapter 4, regarding the system identification of the Honeybee single main rotor helicopter, a full state space model was assembled that captured the 6 DOF motion of the vehicle assuming a nominal trim condition at hover. It was found that cross couplings between many of the states were negligible due to low coherence. Without any loss in accuracy, assuming the physical model in chapter 4, the heave, yaw rotation, and "cyclic" dynamics can be isolated, examined, and controlled independently.

5.0.1 Rotary Wing Cyclic Dynamics Control

The control of the cyclic dynamics is concerned with the roll and pitch angles of the vehicle, as well as the translational motion in the forward and side directions as well. Consider the state space model (5.1), which includes orientation in addition to the states in (4.4). Note again the sources of off-axis coupling. There are *aerodynamic* sources captured by the stability B_a and A_b and there are *gyroscopic* couplings captured in the control sensitivity matrix, B_{lon} and A_{lat}.

$$\begin{Bmatrix} \dot{b} \\ \dot{p} \\ \dot{a} \\ \dot{q} \\ \dot{\phi} \\ \dot{\theta} \end{Bmatrix} = \begin{bmatrix} -\frac{1}{\tau_f} & -1 & \frac{B_a}{\tau_f} & 0 & 0 & 0 \\ L_b & 0 & 0 & 0 & 0 & 0 \\ \frac{A_b}{\tau_f} & 0 & -\frac{1}{\tau_f} & -1 & 0 & 0 \\ 0 & 0 & M_a & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \begin{Bmatrix} b \\ p \\ a \\ q \\ \phi \\ \theta \end{Bmatrix} + \begin{bmatrix} \frac{B_{lat}}{\tau_f} & \frac{B_{lon}}{\tau_f} \\ 0 & 0 \\ \frac{A_{lat}}{\tau_f} & \frac{A_{lon}}{\tau_f} \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{Bmatrix} \mu_{lat} \\ \mu_{lon} \end{Bmatrix} \quad (5.1)$$

For illustrative purposes, consider only the transfer function for roll angle from roll input given a 1 degree of freedom assumption,

$$\frac{\phi(s)}{\mu_{lat}(s)} = \frac{(B_{lat}L_b/\tau_f)\mu_{lat}(s)}{s(s^2 + \frac{1}{\tau_f}s + L_b)} \quad (5.2)$$

From this transfer function, if the control law includes rate feedback, it can be detrimental to transient performance of the roll angle. Instead, it is best if the effective rotor time constant can be reduced. Small weights are added to the vehicle's Hiller-style paddles, thus increasing rotational inertia, to slow the response of the rotor blade and make the vehicle more pilotable for a human controller. With feedback, this demonstrates that it is best to move the weights inward, or remove them all together to improve the response time of the orientation dynamics without causing unacceptable levels of overshoot.

In the limit, where the rotor time constant is small, the pilot input almost instantaneously becomes a rate response, thus reducing (5.2) to a second order response from a third order.

It was found that a parsimonious yet highly effective controller for the orientation of the Honeybee single main rotor vehicle was a simple *proportional* controller. Feedback tends to obscure the cross-coupling inherent in the vehicle dynamics. For the applications the Honeybee was used for, simple independent loops were sufficient to control the vehicle. The quadrotor vehicle includes single DOF on-board attitude control loops.

Regardless of the vehicle, the reduced dynamic models for translational motion in the longitudinal and lateral degrees of freedom allow the inputs to the vehicle to be reference roll and pitch angles and not raw actuator inputs. This greatly improves vehicle stability and allows any translational controller to assume only a first order dynamic model for translational motion.

5.0.2 Heave/Height Control

Heave velocity and height control was implemented for each test vehicle to restrict the vehicle to planar motions. The Honeybee vehicle is slightly more complicated in that there is a distinct transfer function between the throttle input to the main rotor speed due to the larger rotational inertia of the rotor system. The expected transfer function between a change in rotor speed, or collective speed in the case of the quadrotor, and heave velocity is present in both vehicles.

For altitude control, a proportional/derivative controller was implemented on both the Honeybee and Quadrotor vehicles. The Vicon^{TM} system was able to provide the inertial height estimate and indirectly the heave velocity measurement success-

fully. It was desired that the vehicle be capable of height control without requiring access to off-board estimates, however. The complimentary filtered heave velocity estimate discussed in chapter 3 was used to provide the required estimate. This was used in conjunction with the sonar height estimate to create a P-D controller.

5.0.3 Yaw Control

Yaw control on the Honeybee helicopter was implemented using simple yaw rate gyro feedback. This slowed down the yaw dynamics sufficiently for either the Vicon^{TM} tracking or visual control to take over. The Vicon^{TM} system was used when necessary to provide an absolute heading reference.

5.0.4 Translational Control using Ventral Optic Flow

Availability of ventral optic flow provides a source of feedback for translational motion control. The concept of utilizing a height sensor and ventral optic flow was presented in chapter 3.

Consider the transfer function from roll angle as an input to lateral translational velocity,

$$\frac{v(s)}{\phi(s)} = \frac{g}{s + Y_v}. \tag{5.3}$$

If optic flow is fed back directly, then the effective velocity gain is height dependent, with the effective gain decreasing as the vehicle increases in height. In theory, this gain change is not detrimental to stability, however in practice, pure

delay, noise, and structured uncertainty in the removal of the rotational optic flow limits how low the vehicle can be if optic flow is fed back unscaled. With scaling using a height estimate, the desired gain can be set, resulting in consistent damping performance. The optic flow estimate has also been implemented in tracking configuration, such that the vehicle moves in such a way as to attempt to tracking a desired velocity set-point.

At this time, while both approaches have been implemented, the ventral VLSI optic flow sensor was highly susceptible to the spatial frequency of the floor, thus limiting practical usefulness relative to the velocity control the Vicon^{TM}can provide.

5.1 Summary of Reduced Order Dynamics

Assuming orientation feedback implemented with sufficiently high bandwidth, and it can be assumed that the dynamics of the translational degrees of freedom are sufficiently slow relative to the orientation bandwidth, then a set of *effective* dynamics can be assumed while attempting to utilize vision for the translational dynamics. The quadrotor vehicle bandwidth for these effective dynamics is significantly higher than the Honeybee vehicle due to inherent vehicle design and more advanced actuators. A visual navigation loop operating on the effective orientation dynamics represents a *cascaded* control loop architecture.

The effective translational motion dynamics assumed for the implementation of visual navigation are

$$\dot{v} = Y_v v + g\phi_r \tag{5.4}$$

$$\dot{u} = X_u u - g\theta_r. \tag{5.5}$$

$$\tag{5.6}$$

The effective yaw dynamic model for both the Honeybee and quadrotor vehicle is

$$\dot{r} = N_r r + N_{q_{yaw}} \mu_{yaw}, \tag{5.7}$$

although with differing constants.

In most testing scenarios, it was desired to keep the height constant to allow the assumption of planar motion, thus simplifying implementation. The reduced dynamics for heave motion were considered negligible in this case as the vehicle maintained a constant altitude.

Chapter 6

Implementation of Wide Field Integration for Navigation

The feedback control methodologies presented in the previous chapter are focused only on vehicle stability augmentation. Furthermore, the inertial sensing and ventral optic flow sensing capabilities presented in chapter 3 do not provide any information regarding objects surrounding the vehicle. The ability to navigate through an environment requires capabilities for detecting pose relative to obstacles. The benefits of using optic flow for proximity detection, and the implementation of optic flow processing in VLSI circuitry, was established in chapter 2. This chapter presents the implementation of optic flow-based navigation capabilities that are inspired by the neural architecture of insects.

Flying insects are capable of robust navigation in uncertain environments without the computational complexity that current machine vision algorithms require to perform the same tasks. Recent studies at the behavioral and neurophysiological levels have provided insight on the neural mechanisms that give rise to these sophisticated capabilities. Insects encode optic flow [12], the characteristic patterns of visual motion that form on their retinas as they move. Specialized neurons called tangential cells [13], parse these complex optical flow patterns over large swaths of the visual field to extract visual cues for navigation. The remarkable thing is that insects are capable of executing visual-based behavior with direct sensory feedback,

i.e., without the use of observer-based estimation schemes.

The discovery of honeybee-inspired optic flow 'heuristics' by M. V. Srinivasan [44, 45] has spawned several approaches for local navigation [46, 47, 48, 49, 50, 51, 52], speed control [53, 54, 55, 56], and landing [57, 58]. Additionally, behavioral observations of expansion cues in fruit flies [59] have inspired reflexive obstacle avoidance strategies [60, 61, 62].

Insects mitigate the inherent noise of their local optic flow computations by implementing a form of weighted spatial summing in their lobula plate tangential cells to fuse the many optic flow estimates available over large swaths of their retina [63, 64, 65]. Control of locomotion is achieved via feedback of tangential cell outputs to the flight motor [66]. In doing so, they extract the *relative* information contained in optic flow for control and do not deal with absolute quantities. This is a promising approach for implementation in small vehicles as they typically have limited processing capabilities. In fact, it has been shown that a control system designed around this methodology can completely eliminate the need for dynamic estimation of absolute quantities by utilizing output feedback of optic flow [67, 68, 69].

Insects utilize tangential cells to extract relevant information about the environment by applying weighting functions to the observed optic flow field. Central to the power of this method is that control computations for the insect as well as the vehicle form a tight loop that relies only on these spatial weighting functions with no state estimator dynamics. The lack of a traditional Kalman filter or other state observer greatly simplifies the control computations and allows a higher update rate. The cost is simply that all velocity and positioning estimates are relative and there-

fore result in a control theoretic architecture with moving poles. The benefit is that as long as the movable poles remain stable, the benefit in terms of computational burden outweighs the downsides.

The following chapter utilizes a mathematical formalism called *Wide-Field Integration* [69], which allows the extraction of relative state information from spatially weighted optic flow patterns. This framework is such that tools from control theory can be utilized to prove local stability of the resulting closed loop system.

The implementation described herein assumes availability of optic flow measurements on the azimuthal axis of the vehicle. It is shown that weighted outputs of the resulting spatially periodic optic flow signal are sufficient to enable a flying micro helicopter to negotiate a corridor while the optic flow feedback controls its orientation and lateral position. This methodology is designed to closely mimic the spatial decompositions of the sensed motion field performed within the insect visuomotor system by tangential cells. Implementations were performed for both the Honeybee vehicle and the Quadrotor vehicle, with results from each presented.

6.0.1 Planar Optic Flow Model

To facilitate the development of the spatial weighting functions and control methodology that follows, the optic flow field and the spatial distribution of objects in the environment are modeled as continuous functions of the body-frame-referred viewing angle γ. The nearness function is defined as

$$\mu(\gamma, \mathbf{x}) \quad = \quad \frac{1}{d(\gamma, \mathbf{x})}, \tag{6.1}$$

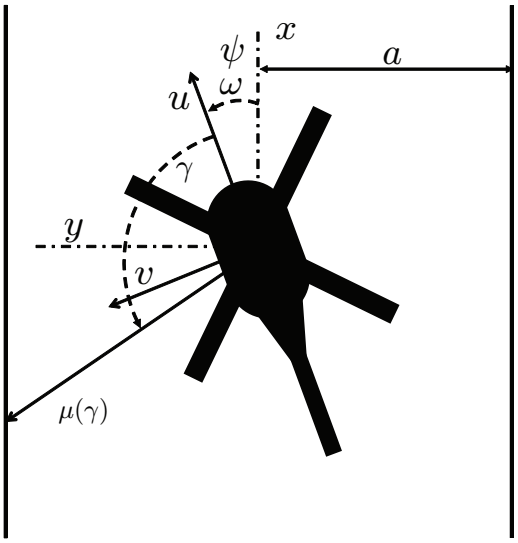

Figure 6.1: Planar Tunnel Coordinate Definitions.

where $d(\gamma, \mathbf{x})$ is a continuous representation of distance to the nearest point in the visual field and $\mathbf{x} = (x, y, \psi)$ is the pose of the vehicle within the environment. The nearness function is assumed to be a bounded, piecewise continuous function with a finite number of discontinuities.

Given this definition, the tangential component of the optic flow on a circular-shaped sensor that is constrained to 3-DOF motion in the horizontal plane can be expressed as a 2π-periodic function in the viewing angle γ:

$$\xi(\gamma) \;=\; -\omega + \mu(\gamma, \mathbf{x})\,(u \sin\gamma - v\cos\gamma). \tag{6.2}$$

6.0.2 Wide-Field Integration

This section summarizes the wide-field integration theory developed by Humbert[67, 68, 69]. Given the tangential component of optic flow (6.2) as a function of the body-fixed viewing angle γ, a methodology is required to interpret the patterns of optic

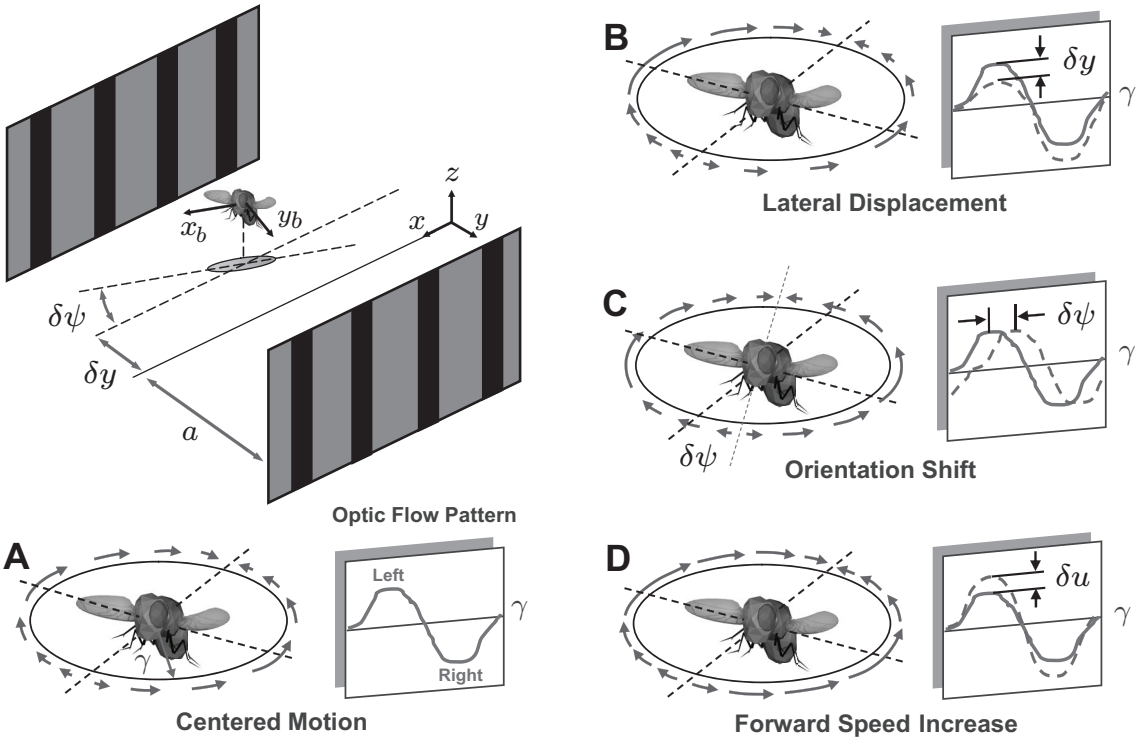

Figure 6.2: Perturbations of azimuthal optic flow and their correlations to the relative state of the vehicle. Amplitude, phase, and asymmetry of the azimuthal optic flow pattern encode relative proximity and speed with respect to obstacles in the environment.

flow that arise during motion. The structure of the visuomotor pathway of insects provides inspiration for a method of extracting relevant information from patterns of optic flow. Tangential cells respond with graded membrane potentials whose polarity depends on the direction of motion [63, 64]. Essentially, the output is a *comparison* between its preferred wide-field pattern of motion and that of the visual stimulus. Mathematically, this process can be represented as an *inner product* $\langle f, g \rangle$, analogous to the dot product between vectors, which tells us how similar two objects f and g are.

The intuition behind this approach is shown in Figure 6.2. Forward motion of a vehicle constrained to move in three degrees of freedom (forward and lateral translation along with yaw rotation) in the horizontal plane generates an optic flow pattern with a focus of expansion in the front field of view, a focus of contraction in the rear, with the largest motion on the sides. If plotted as a function of the angle γ around the retina, this is approximately a sine wave (Figure 6.2A). Perturbations from this equilibrium state of a constant forward velocity u_0 along the centerline of the tunnel introduce either an *asymmetry* in this signal for lateral displacements δy (Figure 6.2B), or a *phase shift* for rotary displacements $\delta\psi$ (Figure 6.2C). If the forward speed is increased by δu, the *amplitude* of this signal increases (Figure 6.2D), and if the vehicle is rotating at angular velocity $r = \dot{\psi}$ about the yaw axis a DC shift in the signal of equal magnitude occurs. Therefore, the amplitude, phase, and asymmetry of the pattern of optic flow around the yaw axis encode important information that could be used for navigation and speed regulation.

In the case of Figure 6.2, the patterns are assumed to reside in $L_2[0, 2\pi]$, the

space of $2\pi-$periodic and square integrable functions, where the inner product is given by

$$z(\mathbf{x}) \;=\; \langle \xi, F \rangle \;=\; \frac{1}{\pi} \int_0^{2\pi} \xi(\gamma, \mathbf{x}) \cdot F(\gamma)\, d\gamma. \tag{6.3}$$

Here $\xi(\gamma, \mathbf{x})$, given in (6.2), is the measured optic flow about the yaw axis, $F(\gamma)$ is any square integrable weighting function such that (6.3) exists, and $z(\mathbf{x})$ is the resulting output which is a function of the state $\mathbf{x}$ of the insect with respect to the environment. This expression, which would represent either a shift in membrane potential or a change in spiking frequency, gives a number that is maximum if the patterns line up, negative if the pattern has the same structure but is in the opposite direction, and zero if the patterns are completely independent (orthogonal) of one another. Hence, the resulting set of outputs generated by a set of weighting functions represent a decomposition of the motion field into simpler pieces that encode perturbations from the desired pattern.

6.1 Navigation of Honeybee Vehicle

This section describes the implementation of the WFI theory on the single main rotor Honeybee vehicle. A state space model of the helicopter dynamics, linearized about the hover condition, has been presented in chapter 4. This identified model confirms the lateral and longitudinal dynamics are highly coupled and high order. Additionally, the heading degree of freedom is highly sensitive to the yaw input and disturbances due to the low rotational damping and inertia. This degree

of freedom is modified via on-board integrated yaw rate feedback to add additional damping. While the helicopter is clearly not a planar constrained vehicle in general, given a fixed altitude and non-aggressive maneuvers, a planar analysis and implementation can be assumed.

The Vicon^{TM}visual tracking system provides an off-board feedback capability that reduces the effective dynamics of the vehicle. Direct measurements of the vehicle orientation, position, and velocity are available at 350 Hz at 10 ms latency. Attitude feedback for the lateral and longitudinal, roll and pitch, degrees of freedom greatly reduces the cross coupling effects inherent in the dynamics, thus allowing the lateral and longitudinal degrees of freedom to be considered separately, and modeled by the set of reduced order dynamics given by

$$\dot{v} = Y_v v + g\phi_{ref} \tag{6.4}$$

$$\dot{u} = X_u u - g\theta_{ref}, \tag{6.5}$$

where Y_v and X_u are aerodynamic damping derivatives, g is gravity, and ϕ_{ref} and θ_{ref} are the reference orientation values for roll and pitch, respectively, which can be considered to be the same as the actual roll and pitch values, ϕ_a and θ_a, for low frequencies. The corresponding transfer function form, given only for the roll degree of freedom, is

$$\frac{v(s)}{\phi_{ref}} = \frac{g}{s + Y_v} \tag{6.6}$$

$$\tag{6.7}$$

These effective lateral and longitudinal transfer functions can be further modified via a simple proportional velocity tracking feedback, given a control gain K_v and a desired velocity v_{ref}, resulting in a transfer function of the following form for the roll degree of freedom:

$$\frac{v_a}{v_{ref}} = \frac{K_v g}{s + (Y_v + K_v g)} \tag{6.8}$$

$$\tag{6.9}$$

A transfer function of identical form is valid for the longitudinal degree of freedom. The velocity tracking loop further reduces the effect of the cross coupling in the vehicle dynamics and adds effective damping. The optic flow derived lateral control operates on the lateral desired commanded velocity. The forward commanded velocity is fixed to a constant value.

Heading control is also augmented using the visual tracking system to supply rate tracking capability. The open loop heading dynamics are given as

$$\dot{r} = N_r r + N_{\mu_y} \mu_y, \tag{6.10}$$

$$\tag{6.11}$$

where r is the yaw rate, N_r is the yaw damping, $N_{\mu y}$ is the control sensitivity, and μ_y is the tail motor input. The term N_r is naturally aerodynamic in origin, however we include in this term the effects of the on-board yaw rate damping as well. Simple proportional feedback is again used to provide yaw rate tracking capability,

resulting in the following transfer function, from desired reference rate, r_{ref} to the actual rate, r_a :

$$\frac{r_a}{r_{ref}} = \frac{K_r N_{\mu_y}}{s + (N_r + K_r N_{\mu_y})} \tag{6.12}$$

$$\tag{6.13}$$

The optic flow derived rate control operates on r_{ref}.

6.1.1 Control Methodology: Honeybee Vehicle

For experimental implementation, we consider the task of navigating a corridor, thus linking perturbation of Fourier harmonics to deviations in desired position and orientation within the tunnel

For the case of a straight corridor, the nearness function $\mu(\gamma, \mathbf{x})$ is independent of the axial position x and can be expressed in closed form as a function of the lateral position y, body frame orientation ψ, and the tunnel half-width a:

$$\mu(\gamma, \mathbf{x}) = \begin{cases} \frac{\sin(\gamma+\psi)}{a-y} & 0 \leq \gamma + \psi < \pi \\ -\frac{\sin(\gamma+\psi)}{a+y} & \pi \leq \gamma + \psi < 2\pi \end{cases} . \tag{6.14}$$

The dynamic model of the vehicle's motion is provided in the body frame of reference, however navigation through a corridor requires consideration of the inertial frame of reference. We denote x and y as the longitudinal and lateral tunnel positions, respectively, and denote the heading orientation as ψ. The equivalent reduced order helicopter dynamic equations for the lateral and rotation degrees of freedom in the inertial frame of reference are:

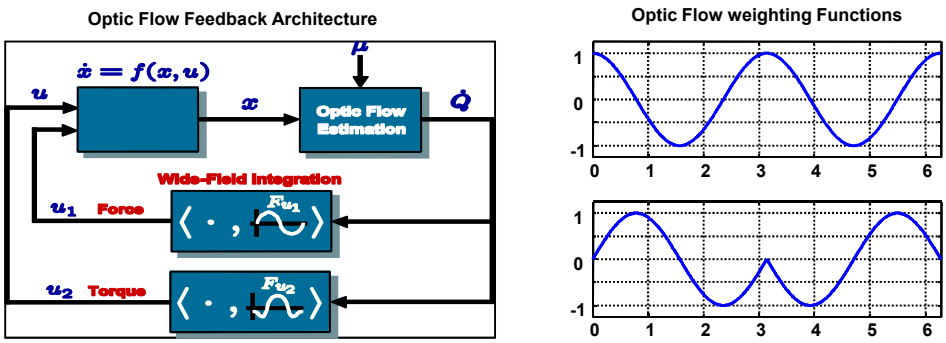

Figure 6.3: Optic Flow Feedback Architecture

$$\ddot{y} = -(Y_v + K_v g)v_a + K_v g v_{ref} \tag{6.15}$$

$$\ddot{\psi} = -(N_r + K_r N_{\mu_y})\dot{\psi} + K_r N_{\mu_y} r_{ref} \tag{6.16}$$

Outputs suitable for feedback are computed using equation 6.3, with weighting functions that extract the relative lateral position and orientation in the tunnel. Lateral position in the tunnel is obtained using the weighting function $F(\gamma) = \cos 2\gamma$. Orientation can be extracted using the weighting function $F(\gamma) = \sin 2\gamma$ for $0 <= \gamma < \pi$ and $F(\gamma) = -\sin 2\gamma$ for $\pi <= \gamma < 2\pi$.

The analytical result of this spatial inner product is nonlinear, however each inner product can be linearized assuming the vehicle is in the desired center lateral position in the tunnel with heading directed straight down the tunnel. Doing so results in the linearized outputs of $z_1 = -\frac{u_{ref}}{2a^2}y$ and $z_2 = -\frac{1}{a}(u_{ref}\psi + \frac{1}{2}\dot{y})$, where u_{ref} is the fixed forward velocity. The motivation for each weighting function is now apparent as the linearized outputs are linear functions of the desired quantities we wish to control. Including these outputs, with arbitrary gains K_y for lateral and K_ψ

for orientation, the closed loop state space model can be computed as:

$$\begin{Bmatrix} \dot{y} \\ \ddot{y} \\ \dot{\psi} \\ \ddot{\psi} \end{Bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -K_y \frac{u_{ref} K_v g}{2a^2} & -(Y_v + K_v g) & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -K_\psi \frac{K_r N_{\mu_y}}{2a} & -K_\psi \frac{u_{ref} K_r N_{\mu_y}}{a} & -(N_r + N_{\mu_y} K_r) \end{bmatrix} \begin{Bmatrix} y \\ \dot{y} \\ \psi \\ \dot{\psi} \end{Bmatrix} \tag{6.17}$$

The Fourier coefficient feedback gains can be adjusted to position closed loop eigenvalues for desired stability.

6.1.2 Results: Honeybee Vehicle

A test utilizing a bent tunnel was conducted to demonstrate the wide-field integration-based feedback on the micro helicopter. A representative trajectory is shown in figure 6.4. The heading of the vehicle at various points on the trajectory is given by the L-shaped markers. This test with the micro helicopter, using noisy sensors on a heavily vibrating platform, demonstrates the potential for using the wide-field integration architecture to fuse relatively noisy optic flow estimates to provide feedback outputs sufficient for navigation.

6.2 Navigation of Quadrotor Vehicle

This section describes the implementation of the WFI theory on the quadrotor vehicle. Emphasis in this case, was placed on eliminating off-board feedback control. Testing of the optic flow navigation method was performed on the X-UFO

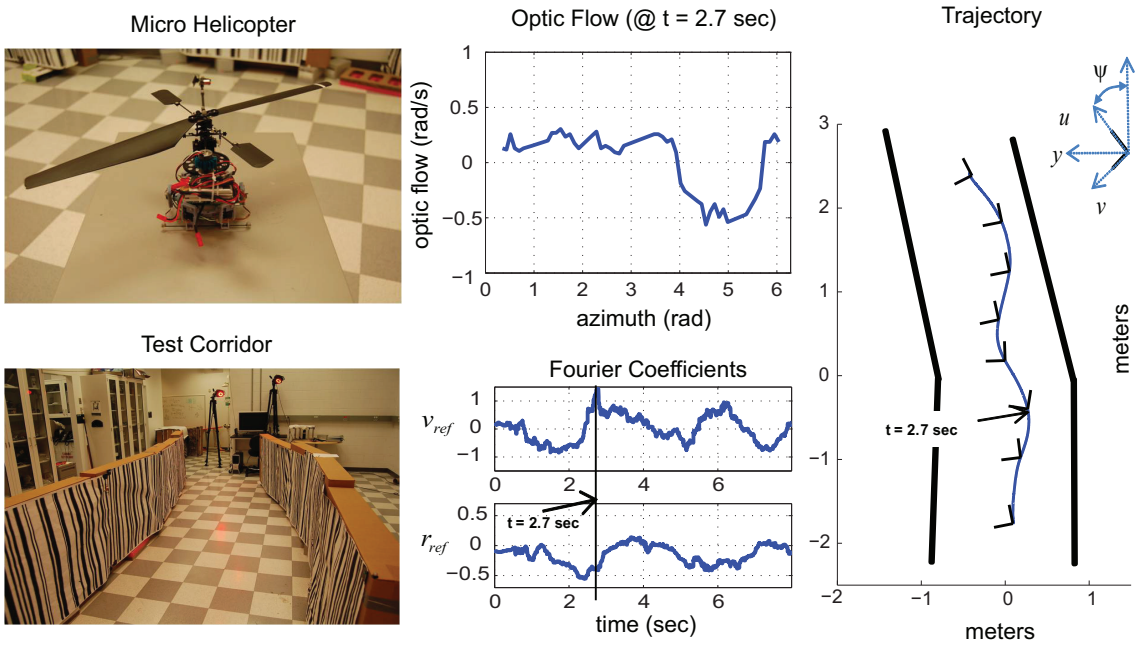

Figure 6.4: Micro Helicopter Flight Trajectory

quadrotor, Figure 1.3, which has a gross weight of 680 grams. The lateral damping was performed with a single ARZ-Lite VLSI optic flow sensor from Centeye, Inc^{TM}. The altitude regulation was accomplished using an onboard Devantech SRF08 ultrasonic range finder. The reference altitude was set to $h_0 = 0.5$ meters. Both of these sensors were located on the underside of the vehicle, facing downward. The in-house avionics consisted of two boards with yaw rate gyros and Bluetooth uplinks. These boards send outer-loop control commands to the X-UFO avionics boards. The X-UFO avionics board(X-BASE) performed attitude feedback stabilization (Figure 2.7).

$$\begin{Bmatrix} \dot{u} \\ \dot{v} \\ \dot{\phi} \\ \dot{\theta} \end{Bmatrix} = \begin{bmatrix} X_u & 0 & 0 & -g \\ 0 & Y_v & g & 0 \\ 0 & 0 & L_\phi & 0 \\ 0 & 0 & 0 & M_\theta \end{bmatrix} \begin{Bmatrix} u \\ v \\ \phi \\ \theta \end{Bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ L_{lat} & 0 \\ 0 & M_{lon} \end{bmatrix} \begin{Bmatrix} \mu_{lat} \\ \mu_{lon} \end{Bmatrix} \tag{6.18}$$

A state space model for the quadrotor was obtained using system identification methods used previously [70], as well as prediction error minimization techniques[71]. The lateral, longitudinal, heave, and yaw dynamics are considered as decoupled, independent systems. The structure of the longitudinal and lateral model is given in (6.18). The dynamic model for the roll and pitch orientations of the vehicle to translational velocity are modeled as first order transfer functions. This set of dynamics is given by

$$\begin{aligned} \dot{u} &= X_u u - g\theta_r, \\ \dot{v} &= Y_v v + g\phi_r, \end{aligned} \tag{6.19}$$

where Y_v and X_u are aerodynamic damping derivatives, g is gravity, and ϕ_r and θ_r are the commanded orientation values for roll and pitch, respectively. System identification of the quadrotor vehicle dynamics suggests that for low frequencies the commanded roll and pitch angles can be considered the same as the actual angles, ϕ and θ.

The lateral and longitudinal dynamics were augmented via proportional velocity damping feedback using averaged translational optic flow estimates $\xi_u = u/h_0$ and $\xi_v = v/h_0$ in the u and v directions, respectively. The rotational component

of optic flow provided by the ventrally located Centeye sensor was removed from these measurements using the on-board roll and pitch rate gyro estimates. Given a control gain K_v, the resulting transfer function for the lateral degree of freedom is

$$\frac{v}{\phi_r} = \frac{g}{s + (Y_v + \frac{K_v}{h_0})}. \tag{6.20}$$

A transfer function of identical form is valid for the longitudinal degree of freedom, using a longitudinal velocity feedback gain, K_u.

The dynamic model of the heading degree of freedom is also modeled as a first order system. The on-board avionics provide yaw rate tracking such that the open loop heading dynamics are given as

$$\dot{r} = -(N_r + K_r)r + N_{\mu_{yaw}}\mu_{yaw}, \tag{6.21}$$

where r is the yaw rate, N_r is the yaw damping, $N_{\mu_{yaw}}$ is the control sensitivity, and μ_{yaw} is the yaw input. The term N_r in this case determines the time constant of the closed loop tracking bandwidth as provided by the on-board avionics.

The heave dynamics are regulated via the fusion of a sonar ranging sensor and a vertically oriented accelerometer. The transfer function for the heave dynamics was identified as a first order system,

$$\dot{w} = Z_w w + Z_{\mu_{th}}\mu_{th}, \tag{6.22}$$

and altitude control is implemented as a PD feedback control loop. The heave velocity estimate, w, required for the derivative feedback, was estimated via a complimentary filtering algorithm using the sonar and accelerometer as presented in [35]. The desired height reference for testing was set to a constant 0.5 meters.

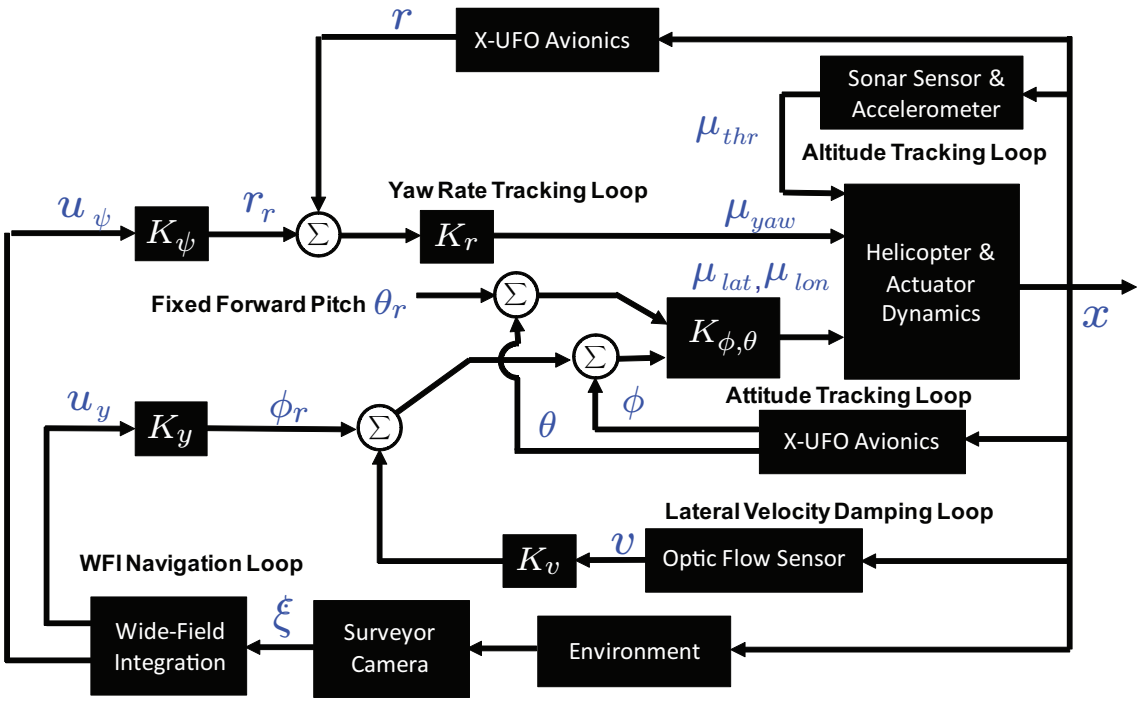

Figure 6.5: Closed loop block diagram for WFI-based navigation. Inner loop pitch and roll control is accomplished using onboard gyro-based avionics while optic flow derived outputs are used to close the outer navigation loop.

Figure 6.5 shows the overall control architecture that was employed, including the inner stabilization loops and outer visual navigation loops. The gains K_r, K_ϕ and K_θ are adjustable, but internal to the X-UFO avionics.

6.2.1 Control Methodology: Quadrotor Vehicle

In this section, we describe the outer loop control methodology that was implemented to achieve navigation of a straight corridor. The above lateral (6.20) and yaw (6.21) dynamics are provided in the body frame of reference. To facilitate the selection of the spatial weighting functions, we express these equations, along with (6.2), in terms of the pose of the vehicle $\mathbf{x} = (x, y, \psi)$, where x and y denote the longitudinal and lateral positions relative to the corridor, respectively, and the heading orientation is denoted as ψ. The equivalent inner-loop, reduced order vehicle dynamics for the lateral and yaw degrees of freedom in the inertial frame of reference are

$$\begin{aligned} \ddot{x} &= -(X_u + \frac{K_u}{h_0})\dot{x} + g\theta_r \\ \ddot{y} &= -(Y_v + \frac{K_v}{h_0})\dot{y} + g\phi_r \\ \ddot{\psi} &= -(N_r + K_r)\dot{\psi} + N_{\mu_y} r_r. \end{aligned} \tag{6.23}$$

Given that the altitude is held constant, the desired roll angle, ϕ_r, pitch angle, θ_r, and yaw rate input r_r are available to control the motion of the vehicle through the tunnel environment. The pitch input is set to a constant value to induce a fixed forward flight speed. Control commands are computed from the wide-field integrated optic flow field and used directly as inputs to the lateral and heading

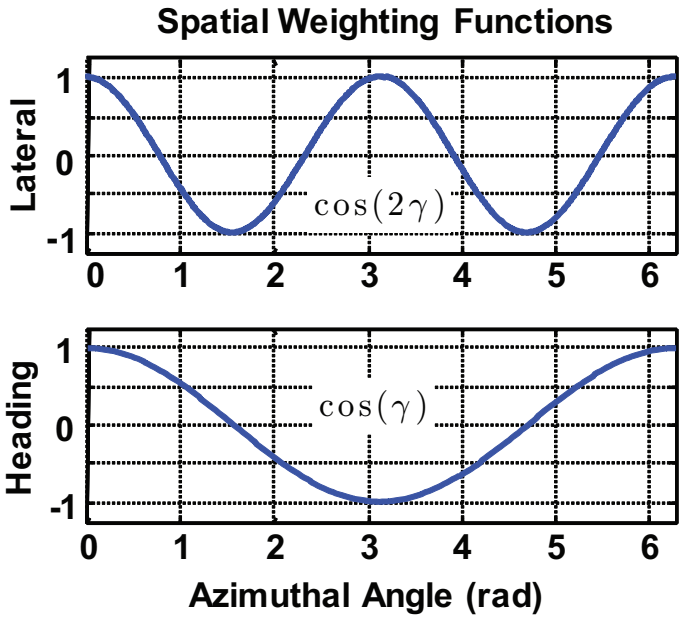

Figure 6.6: Spatial Weighting Functions.

inputs.

Outputs suitable for outer-loop feedback commands ϕ_r and r_r are computed using (6.3), with weighting functions that extract the relative lateral position and orientation in the corridor. The output proportional to the relative lateral position in the corridor, z_1, is obtained using the weighting function $F_y(\gamma) = \cos 2\gamma$, and the output that contains the orientation relative to the tunnel centerline, z_2, can be extracted using the weighting function $F_\psi(\gamma) = \cos\gamma$. These weighting functions are shown in Figure 6.6. For our analytical model of optic flow (6.2), the spatial inner products expressed in inertial coordinates can be expressed as

$$\begin{aligned} z_1 &= \langle \xi, F_y \rangle &&= -\frac{y}{2(a^2 - y^2)}(\dot{x}\cos 2\psi + \dot{y}\sin 2\psi) \\ z_2 &= \langle \xi, F_\psi \rangle &&= \frac{4a}{3\pi(a^2 - y^2)}(2\dot{x}\cos\psi + \dot{y}\sin\psi). \end{aligned} \tag{6.24}$$

The equilibrium condition $\mathbf{x}_0$ for centering behavior is that the vehicle is posi-

tioned at a height $h = h_0$, lateral position $y = 0$, with zero orientation offset, $\psi = 0$, and with a forward velocity, $u = u_0$. The linearized outputs are

$$\begin{aligned} z_1 &= -\frac{u_0}{2a^2} y \\ z_2 &= \frac{1}{a}(u_0\psi - \frac{1}{2}\dot{y}). \end{aligned} \tag{6.25}$$

To close the outer feedback loop that governs heading and lateral control, reference inputs ϕ_r and r_r in (6.23) are set equal to (6.24), scaled by gains K_ψ and K_y:

$$\begin{aligned} \phi_r &= K_y \langle \xi, F_y \rangle \\ r_r &= K_\psi \langle \xi, F_\psi \rangle. \end{aligned} \tag{6.26}$$

$$\begin{Bmatrix} \dot{y} \\ \ddot{y} \\ \dot{\psi} \\ \ddot{\psi} \end{Bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -K_y \frac{u_0 g}{2a^2} & -(Y_v + \frac{K_v}{h_0}) & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & K_\psi \frac{N_{\mu_{yaw}}}{2a} & -K_\psi \frac{u_0 N_{\mu_{yaw}}}{a} & -(N_r + K_r) \end{bmatrix} \begin{Bmatrix} y \\ \dot{y} \\ \psi \\ \dot{\psi} \end{Bmatrix}.$$

The resulting closed loop linearized dynamics are given in (6.27). The feedback gains K_y and K_ψ, which determine the amount of lateral and rotational stiffness added to the dynamics, can be adjusted to position closed loop eigenvalues for desired stability and performance. For this demonstration, gains were selected such that the K_y gain was equal to that of the K_ψ gain.

6.2.2 Results: Quadrotor Vehicle

In this section, the results of the corridor navigation experimentation are presented and interpreted. Using the feedback methodology described above, the

Figure 6.7: Corridor Test Environment.

quadrotor was capable of navigating a corridor-like environment autonomously. The experimentation was performed in a corridor 1.5 meters wide and 9 meters long (Figure 6.7). The procedure for each flight test was to lift off and position the vehicle at its initial flight condition at the beginning of the corridor, engage the sonar-based altitude control, input a step in reference pitch angle θ_r to the vehicle to begin moving down the corridor, and engage optic flow for lateral and orientation control. The vehicle trajectories of six separate flights are shown in Figure 6.8A. To collect this trajectory data, a Vicon^{TM}visual tracking system was implemented. The vehicle clearly avoids collision with the walls. The circular marker in Figure 6.8A indicates the point in one of the trajectories to which the optic flow in Figure 6.8B corresponds. Figure 6.8C displays the time history of the WFI coefficients with the

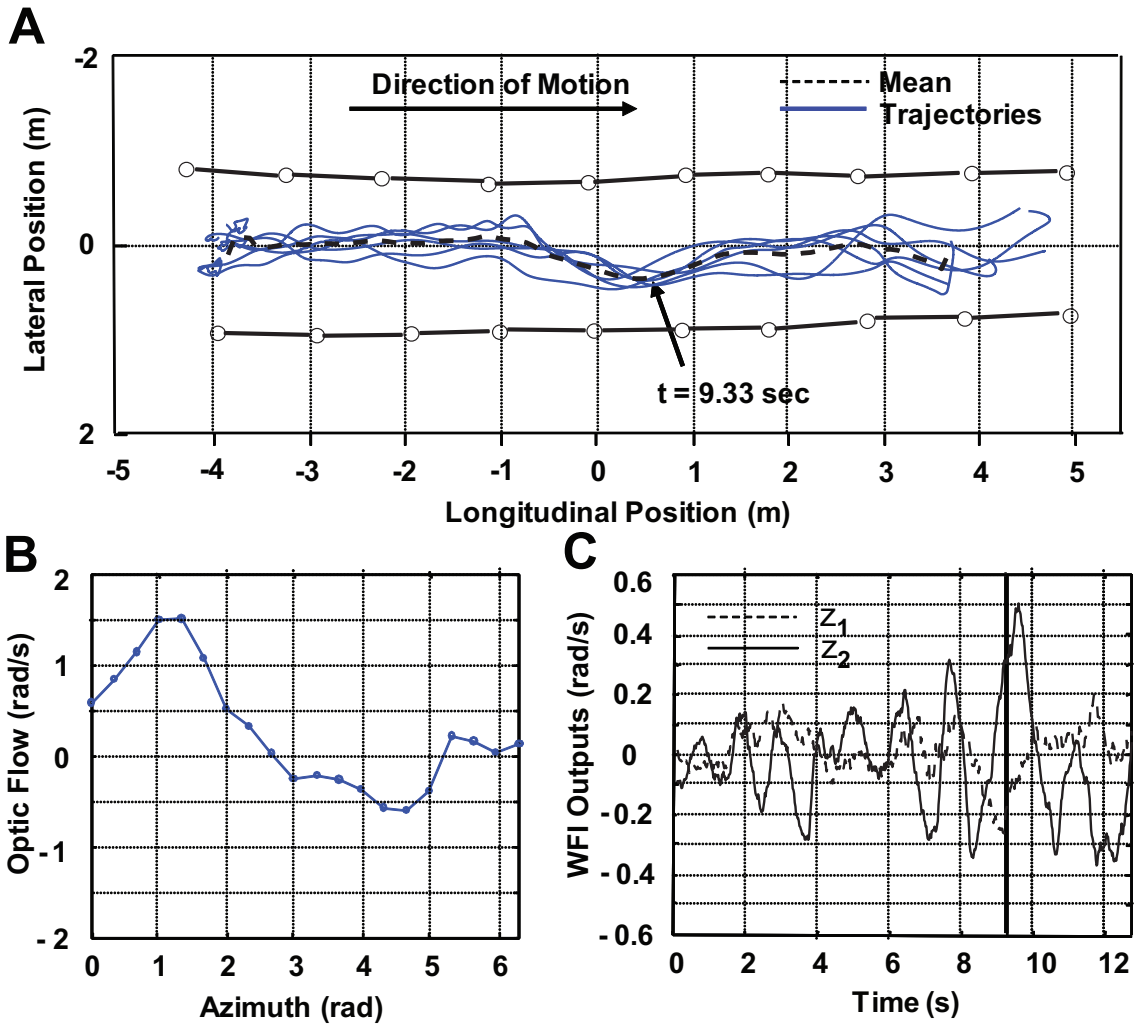

Figure 6.8: Vehicle trajectory through the corridor (A), Optic flow at one instant in time (B), Time history of WFI coefficients.

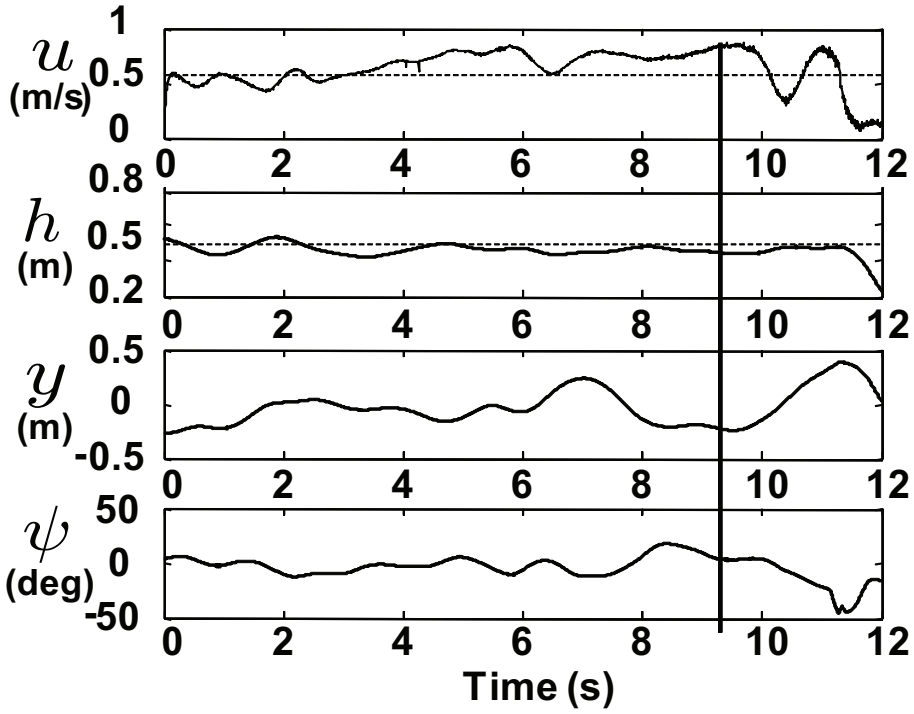

Figure 6.9: Time history of forward velocity, altitude, lateral offset, and orientation.

point corresponding to Figure 6.8B marked by a vertical line. The spatial optic flow distribution demonstrates how the optic flow has an asymmetry as the vehicle drifts too close to one side of the corridor. Figure 6.8C shows the increase in the linearized WFI output at this instant in time, leading to a lateral and orientation input that returns the vehicle to the center of the corridor. The vibrations and orientation changes present on the quadrotor reduce the quality of the optic flow estimates, however as is demonstrated, the wide-field integration methodology is sufficient to affect navigation capabilities, in spite of these difficulties.

Figure 6.9 displays the timetraces of the forward velocity, altitude, lateral offset, and orientation. They indicate that altitude was held constant, and that forward speed changed minimally. This speed change was due to a low forward

speed regulation gain, and turbulent flow in the corridor due to rotor downwash. The vertical line, which indicates the instant in time corresponding to Figure 6.8B, depicts a negative lateral offset and essentially no yaw angle. This corresponds to the trajectory in Figure 6.8A and the WFI coefficients in Figure 6.8C. The optic flow control for orientation and lateral position were disengaged at approximately 10 seconds into the flight test. Due the destabilizing effect of the sideslip on the z_1 output it was required that the heading gain be small. Additional lateral velocity damping would hypothetically allow a higher gain for the z_1 output.

Chapter 7

Nonlinear, Omni-directional Structure from Motion Observer

7.1 Active Vision in Insects

Chapter 6 discussed prior research demonstrating how insects are capable of using optic flow to provide relative motion cues that enable navigation. A convergence layer within the insect brain takes optic flow information available from a wide field of view and produces a small number of outputs that can be fed directly back to the flight motor system for control. Wide field integration of the omni-directional optic flow field was shown to be a mathematical analog to this process of creating a few outputs from a large number of noisy optic flow measurements.

This process inherently favors the low spatial frequency structure, as perceived by the observer, within the environment. That is, large objects are weighted such that they are more important than small objects. Many smaller objects, such as posts or wires, can present a fatal collision hazard just as serious as larger objects can present.

Insects are known to be able to detect, track, and either pursue or avoid conspecifics [72], and can find structural elements that take up a relatively small portion of the overall visual field [73, 74].

One particular insect capability that is of interest for this research is the ability to find structural elements that take up a relatively small portion of the overall visual

field [73, 74]. As they generally lack stereoscopic vision, insects apply *active vision*, the extraction of structure by induced self-motion, to identify these element[75]. For instance, members of a particular wasp species have been shown to leave their nest, memorize the structure of the area immediately surrounding their nest, and be able to easily find their nest again when they return by executing a few simple flight maneuvers. Figure 7.1, reproduced from [75], shows example flight paths flown by the wasp species *Odynerus spinipes* when investigating a small styrofoam ball.

For instance, members of a particular wasp species have been shown to leave their nest, memorize the structure of the area immediately surrounding their nest, and be able to find their nest again when they return by extracting the structure.

Insects are not capable of stereo vision in general to discriminate structure. Instead, they perform flight maneuvers to generate optic flow, and from the structural information comes from relative magnitudes and local changes in the optic flow field. The process whereby insects utilize self motion to extract structural information is known as *active vision*[75]

This chapter presents a dynamic observer that is designed to emulate the insects ability to extract structure by exploiting active vision. The design leverages the continuous-time processing and near omni-directional visual field exhibited by insects to produce a detailed depth map of an arbitrary environment.

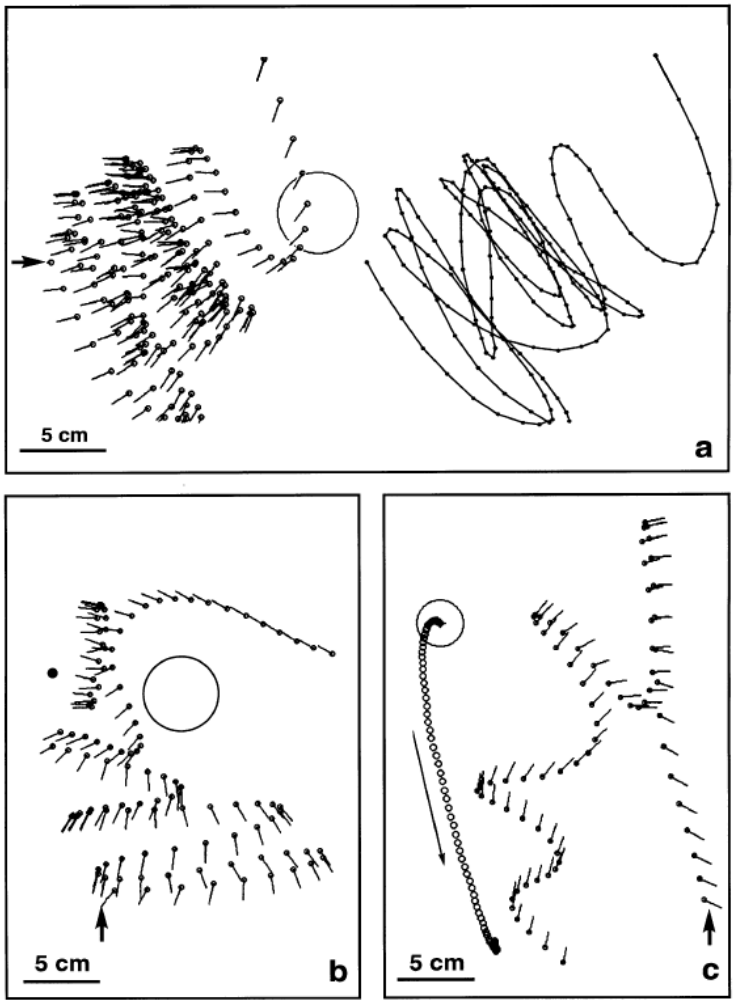

Figure 7.1: a-c: The zig-zagging flight of Odynerus spinipes. a. A flight directed towards a black styrofoam ball as seen from above. b. Different wasp flight path; nest entrance denoted by black dot. c. Motion directed towards a moving object.[75]

7.2 Structure from Motion Literature Survey

An attempt to extract environmental structure from either tracked features or optic flow in the computer vision community is typically referred to as *structure from motion*. The concept of extracting structure from a series of images began in the early 1980's[76, 77, 78]. Typically, structure from motion is broken into

two distinct procedures. First, a set of distinct features must be identified in two frames and their displacements identified. This is usually referred to as solving the *correspondence problem*. Secondly, the structure of the environment is extracted from the list of features. Earlier developments focused on the usage of imagery with typical angular fields of view (∼40-50 deg diagonal) and considering two images at once[79, 80, 81, 82, 83, 84]. It is possible to estimate structure from motion without directly identifying correspondence points by computing optic flow of the image sequence. Optic flow will generally result in more densely packed motion estimates, at equidistant intervals, however since these algorithms estimate spatio-temporal gradients and don't explicitly attempt to establish correspondence, the motion estimates are less accurate.

While the underlying theory for various structure from motion implementations are related to the same fundamental ideas[85, 86], particular estimation methods can take advantage of additional assumed constraints[87] or varying fields of view[88].

Furthermore, improvements in the structural estimation are possible by using a set of several or more images[89] to compute structure. Processing several images at once, also referred to as *batch processing* would result in a significant delay if the estimate was desired in real-time. Instead, it is possible to formulate structural estimation in a recursive nature, allowing a continuously refined estimate to be available after each new image is considered.

Kalman filtering is an attractive approach for continuously improving an estimate given new information, however the traditional formulation of the Kalman

filter is linear. If the estimation algorithm is linear in nature, then only a smaller field of view is possible, and linear methodologies such as epipolar optimization are inherently unreliable using small fields of view[90].

The traditional Kalman filtering approach can be adapted to handle the inherent nonlinear nature of identifying structure via extended Kalman filtering[91, 92]. These approaches however, lack a complete guarantee of convergence[93].

Nonlinear approaches utilizing Lyapunov stability proofs can guarantee convergence, however an appropriate Lyapunov function must be identified and exponential convergence must be explicitly shown[94].

As they are developed without the smaller field of view assumptions typical of linear estimation strategies, full nonlinear estimation strategies will benefit from maximizing the field of view. An omni-directional visual field has the advantage that tracked features over the image sequence are less likely to result in ambiguities that result from a small field of view[95]. Omni-directional structure from motion can be formulated to use both two-step approaches[96, 97, 98] and recursive approaches[99, 100, 101].

Recursive and omni-directional structure from motion observers utilizing feature-tracking have been recently developed [93, 102]. This particular research focuses on the development of a recursive structure from motion observer utilizing omni-directional availability of optic flow estimates. Formulating the observer utilizing optic flow as the input maintains a plausible connection to biological processes. The continuous-time formulation is appropriate given the continuous-time nature of biological computation and the effective rate at which new motion estimates become

available. The omni-directional field of view best represents the wide-field of view insects exhibit with compound eyes. Insects are not purported to be computing structure from motion when performing maneuvers associated with active vision, however. Instead, this structure from motion observer is designed to capture the effects of the, as yet, unknown process by which insects extract and memorize environmental structure.

7.3 Nonlinear Observer Design

The observer assumes attitude control such that the vehicle motion limits roll and pitch angles and is effectively limited to translation and yaw rotation on the plane normal to gravity. For this reason, the one-dimensional representation for omni-directional optic flow, $\xi(\gamma)$, utilized in the previous research on wide-field integration, is deemed suitable.

$$\xi(\gamma) = -\omega + \mu(\gamma)u\sin(\gamma) - \mu(\gamma)v\cos(\gamma). \tag{7.1}$$

Figure 7.2 shows the chosen coordinate system . The self rotation, ω, and forward/lateral translational velocities, u and v, are assumed to be known, either through a separate measurements or perhaps via a dynamic model of the vehicle dynamics. The nearness $\mu(\gamma)$, a function of viewing angle γ, is the inverse of the current distance to objects in the environment. The notation $\mu(\gamma, \psi, x, y)$ reflects the fact that the nearness depends on pose within the environment.

Defining $\lambda(\gamma) = u\sin(\gamma) - v\cos(\gamma)$ as the *velocity distribution*,

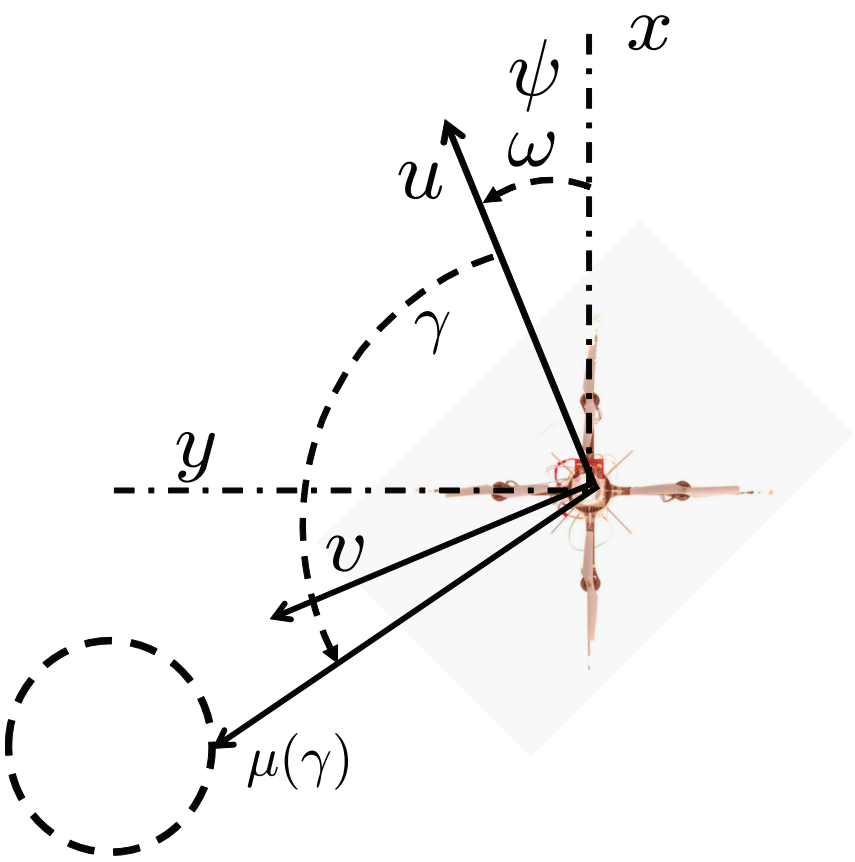

Figure 7.2: Coordinate System Definition for Observer

$$\xi(\gamma) = -\omega + \mu(\gamma)\lambda(\gamma), \tag{7.2}$$

is a convenient method expressing the optic flow for the following analysis. The goal of this observer is to determine a current estimate of $\widehat{\mu}(\gamma)$, given a continuous measurement of $\xi(\gamma)$.

Given $\xi(\gamma)$, the goal is to determine an estimate $\widehat{\mu}(\gamma)$. It is assumed that ω, u and v is known to within some error bound.

A well known problem regarding the simultaneous estimation of structure and motion is that they can only be determined within a scaling constant. Simply by

inspection of equation 7.2, one can see that the same $\xi(\gamma)$ can result from an infinite set of $\mu(\gamma)\lambda$ possibilities. With additional information about either term, however, these terms can be determined uniquely.

7.4 Static Estimation of Structure

Assuming self-motion is known, a static estimate of the structure, $\mu(\gamma)$, can be obtained from equation (7.2) as

$$\mu(\gamma) = (\omega - \xi(\gamma))\frac{1}{\lambda(\gamma)}. \tag{7.3}$$

This estimate must neglect areas where $\lambda(\gamma) = 0$ as the estimate is undefined. Furthermore, for practical consideration, areas where $\xi(\gamma)$ or $\lambda(\gamma)$ are even close to zero will result in noisy structure estimates. It is for this additional reason the problem is formulated in terms of a continuous-time observer which avoids this possibly ill posed inversion.

7.4.1 Proof Assuming no Structured Uncertainty

A Lyapunov proof, demonstrating convergence follows in this section. Given a generally changing retinal pattern of $\xi(\gamma)$, the goal of the observer is to continuously drive the estimated nearness function estimate, $\widehat{\mu}(\gamma)$ towards the true nearness, $\mu(\gamma)$. Formally, the goal is represented as the minimization of the L_2 function norm $\|\mu(\gamma) - \widehat{\mu}(\gamma)\|$. Here, $\xi(\gamma) \in L_2[0, 2\pi]$ and $\mu(\gamma) \in L_2[0, 2\pi]$.

A nonnegative energy function W was chosen to represent the deviation of

$\mu(\gamma)$ from $\widehat{\mu}(\gamma)$ as

$$W = \frac{1}{2\rho}\|\widehat{\mu}(\gamma) - \mu(\gamma)\| = \frac{1}{2\rho}\int_0^{2\pi}(\widehat{\mu}(\gamma) - \mu(\gamma))^2 d\gamma = \frac{1}{2\rho}\int_0^{2\pi}\widetilde{\mu}(\gamma)^2 d\gamma. \tag{7.4}$$

The term ρ is simply a positive scalar value and $\widetilde{\mu}(\gamma)$ is used to denote the difference between the estimate and the actual nearness. A minimum of this energy function is attained if $\dot{W} = 0$, and stability is ensured by ensuring $\dot{W} < 0$. This results in:

$$\dot{W} = \frac{1}{\rho}\int_0^{2\pi}(\widehat{\mu}(\gamma) - \mu(\gamma))(\dot{\widehat{\mu}}(\gamma) - \dot{\mu}(\gamma))d\gamma \tag{7.5}$$

The term $\dot{\widehat{\mu}}(\gamma)$ can be specified such that $\dot{W}$ is ensured negative. One possible choice for $\dot{\widehat{\mu}}(\gamma)$ is

$$\dot{\widehat{\mu}}(\gamma) = \dot{\mu}_m(\gamma) - \rho\widetilde{\xi}(\gamma)\widehat{\lambda}. \tag{7.6}$$

Here, $\dot{\mu}_m(\gamma)$ is included to cancel the effect of how the nearness function is *expected* to change over time, given the current nearness and egomotion.

Substituting equation (7.6) into (7.5) results in

$$\dot{W} = \frac{1}{\rho}\int_0^{2\pi}\widetilde{\mu}(\gamma)(\dot{\mu}_m(\gamma) - \dot{\mu}(\gamma)) - \rho\widetilde{\mu}(\gamma)\widetilde{\xi}(\gamma)\widehat{\lambda}(\gamma)d\gamma \tag{7.7}$$

Assuming $\dot{\mu}_m(\gamma) = \dot{\mu}(\gamma)$, this simplifies to

$$\dot{W} = \frac{1}{\rho}\int_0^{2\pi} -\rho\widetilde{\mu}(\gamma)\widetilde{\xi}(\gamma)\widehat{\lambda}(\gamma)d\gamma \tag{7.8}$$

The integrand contains the term $\widetilde{\xi}(\gamma)$ for which a substitution can be derived from equation 7.2 as follows.

Consider the difference between the estimated and actual optic flow,

$$\widetilde{\xi}(\gamma) = (\widehat{\omega} - \omega) + \widehat{\mu}(\gamma)\widehat{\lambda} - \mu(\gamma)\lambda. \tag{7.9}$$

This can be rearranged as

$$\widetilde{\xi}(\gamma) = \widetilde{\omega} + \widetilde{\mu}(\gamma)\widehat{\lambda} + \mu(\gamma)\widetilde{\lambda}. \tag{7.10}$$

Assume initially that $\widetilde{\omega}$ and $\widetilde{\lambda}$ are zero, indicating perfect knowledge of ω, u, and v. Doing so reduces equation (7.10) to

$$\widetilde{\xi}(\gamma) = \widetilde{\mu}(\gamma)\widehat{\lambda} \tag{7.11}$$

and substitution of this modifies equation (7.8) to become

$$\dot{W} \leq -\int_0^{2\pi} \widehat{\lambda}(\gamma)^2 \widetilde{\mu}(\gamma)^2 d\gamma \tag{7.12}$$

Compare to the original storage function in (7.4) and the similarity is apparent. $\dot{W}$ is the spatial inner product of the two positive functions $\widehat{\lambda}(\gamma)^2$ and $\widetilde{\mu}(\gamma)^2$. If $\widehat{\lambda}(\gamma)^2$ was some positive constant, then exponential stability could be concluded directly. This is not the case in general therefore the assurance of a non-zero $\widehat{\lambda}(\gamma)^2$ term represents a persistency of excitation requirement that must be satisfied.

7.4.2 Requirement for Sinusoidal Motion

Considering equation 7.12, it is at first not apparent why sinusoidal motion is required for examining a particular region of interest. In fact, given motion orthogonal to the desired viewing convergence area, a constant velocity would induce the fastest convergence rate. The reason for this is that the formulation that has been developed does not explicitly account for the effects of inertial location on the nearness function. Consider a point on an obstacle in inertial space that is directly orthogonal to the current motion of the vehicle. As the vehicle moves, the gamma coordinate associated with the object being viewed shifts. Over time, as the inertial point moves to a viewing angle that is closer to the focii of expansion or contraction, the convergence rate drops off. For this reason, if the structure at particular inertial location is to be resolved, the best motion is that which maximizes velocity orthogonal to the viewing angle of the structure and yet attempts to maintain the object at that viewing location.

One might suggest a flight path that encircles the structure, thus allowing both of the above requirements to be met without sinusoidal motion, however a particular structure will not generally elicit the same nearness from all angles. The nearness function associated with a particular obstacle structure will indeed be position dependent and assumed to be unique. One only needs to consider a rectangular obstacle, shown in figure

7.4.3 Persistency of Excitation

Given equation (7.12), it is difficult to separate the $\widehat{\lambda}^2(\gamma)$ term from the integral such that exponential convergence can be shown in a spatially global sense. It is certainly true that

$$\dot{W} \leq -\min_{\gamma}\widehat{\lambda}(\gamma)^2 \int_0^{2\pi} \widetilde{\mu}(\gamma)^2 d\gamma. \tag{7.13}$$

The square of the velocity distribution, $\lambda(\gamma)$, is

$$\lambda(\gamma)^2 = u^2 \sin^2(\gamma) - 2uv \sin(\gamma)\cos(\gamma) + v^2 \cos^2(\gamma). \tag{7.14}$$

The velocity distribution can be assured to be exactly zero at two locations of γ at all times, however. These two locations are, in fact, the *focus of expansion* and *focus of contraction*, which can be found by solving equation (7.14) for the zero locations:

$$\gamma_0 = \operatorname{atan}\left(\frac{v}{u}\right) \tag{7.15}$$

The presence of these points ensure that $\min_{\gamma}\widehat{\lambda}(\gamma)^2 = 0$ for all time. Given a particular u,v and γ, $\lambda(\gamma)^2$ can be thought of as a gain on the exponential convergence of the estimation error.

If $\lambda(\gamma)^2$ can be shown to be a non-zero, positive constant for all γ, then exponential stability can be inferred directly. This is, however, impossible at any one time. For constant values of u and v, there is no guarantee of convergence for all γ locations. To guarantee exponential stability of the observer, a constraint must

be placed on (7.14), such that the term is never zero at any γ location for more than a short period of time.

To best illustrate the global spatial convergence of the observer, consider (7.13) assuming a piece-wise constant discretized form of the original Lyapunov function, with a time dependent velocity distribution. The discretization is actually appropriate from the standpoint of the practical implementation of the observer. The nearness function estimate is approximated with a finite number of values. The analysis will proceed assuming piecewise-constant discretization:

$$W = \sum_{k=0}^{K-1} W_k; k = 1...K-1 \tag{7.16}$$

$$W_k = \int_{2\pi k/K}^{2\pi(k+1)/K} \widetilde{\mu}(\gamma)^2 d\gamma \tag{7.17}$$

and

$$\dot{W} = \sum_{k=0}^{K-1} \dot{W}_k \tag{7.18}$$

$$\dot{W}_k = -\int_{2\pi k/K}^{2\pi(k+1)/K} \widehat{\lambda}(\gamma,t)^2 \widetilde{\mu}(\gamma)^2 d\gamma \tag{7.19}$$

Define

$$\alpha_k(t) = \min_{\gamma=[\frac{2\pi k}{K}, \frac{2\pi k+1}{K}]} \widehat{\lambda}(\gamma,t)^2. \tag{7.20}$$

Then

$$\dot{W}_k = -\alpha_k(t) \int_{2\pi k/K}^{2\pi(k+1)/K} \widetilde{\mu}(\gamma)^2 d\gamma = -\alpha_k(t) W_k \tag{7.21}$$

The persistency of excitation requires that there exists some finite time, T, such that for every $t > 0$ and every value of k the following condition holds:

$$\int_{t}^{t+T} \alpha_k(\tau) d\tau \geq \alpha_0 > 0. \tag{7.22}$$

Given the discretization of the Lyapunov function, there are two values of k for which α_k will be zero at any given time. An overall *average* exponential convergence can be guaranteed by continually altering $\widehat{\lambda}(\gamma)^2$ such that α_k is only zero for a short time. The shorter time γ_0 is such that α_k is zero at a particular location, the shorter T can be to ensure a positive α_k The alteration of $\widehat{\lambda}(\gamma)^2$ is affected via appropriate $u(t)$ and $v(t)$ designed to alter γ_0 continuously.

7.4.4 Prototypical Flight Paths for Optimal Spatial Observer Convergence

Assuming a particular velocity profile over a set length of time, $\lambda(\gamma, t)^2$, the time integrated velocity profile indicates coverage over all γ locations over a fixed period of time. To optimize the velocity profile for coverage over a period of time, consider the minimization of a cost function of the spatial variance of the integrated velocity profile.

The time integrated velocity profile will be denoted as $\acute{\lambda}(\gamma)$:

$$\acute{\lambda}(\gamma) = \int_{0}^{T} \lambda(\gamma, t)^2 dt \tag{7.23}$$

Minimizing the variance of this would amount to minimizing

$$Var(\acute{\lambda}(\gamma)) = \int_0^{2\pi} (\acute{\lambda} - \nu)^2 d\gamma \tag{7.24}$$

where ν is the spatial mean of $\acute{\lambda}(\gamma)$.

Taking the derivative of the variance and setting to zero to identify the extrema:

$$\int_0^{2\pi} 2(\acute{\lambda}(\gamma) - \nu)\frac{\partial \acute{\lambda}(\gamma)}{\partial \gamma} d\gamma = 0 \tag{7.25}$$

Assuming a fixed forward velocity is desired, it can be immediately seen that $\acute{\lambda}(\gamma)$ will in fact be zero at $\gamma = 0, \pi$, therefore some level of sideslip, v, is required to ensure better coverage over all γ locations. Assume a periodic $v(t) = v_0 \sin(t)$ with the fixed forward velocity $u_0 = 1$, integrating temporally from 0 to 2π (one period) and substituting into equation (7.25).

$$\int_0^{2\pi} 2(4\sin^2(\gamma)\pi + 2v_0^2\cos^2(\gamma)\pi - 2\pi - v_0^2\pi)v_0\cos^2(\gamma)\pi d\gamma = 0 \tag{7.26}$$

Integrating to eliminate the dependence on γ:

$$-2v_0\pi^3 + v_0^3\pi^3 = (v^2 - 2)v = 0 \tag{7.27}$$

This leaves three possibilities for values of v_0 that will produce an extremum in the cost function. The trivial case, $v = 0$, is a worst case which minimizes the coverage of the squared velocity distribution. $v = +/-\sqrt{2}$ maximizes the coverage, and in fact results in a constant value for all γ at the conclusion of each cycle.

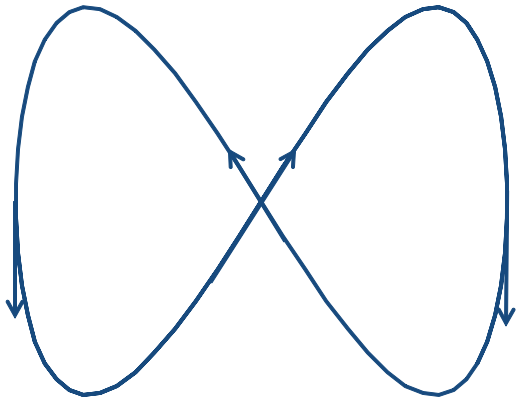

Figure 7.3: Prototypical Bow-tie Flight Pattern for Omni-directional Convergence

Consider a sinusoidal forward velocity profile next, $u = \sin(t)$ and a lateral velocity profile, $v = v_0 \sin(t)$. Minimizing (7.25) assuming this results in:

$$\int_0^{2\pi} 2(2\sin^2(\gamma)\pi + 2v_0^2\cos^2(\gamma)\pi - \pi - v_0^2\pi)v_0\cos^2(\gamma)\pi d\gamma = 0 \tag{7.28}$$

Neglecting the trivial minimum, $v_0 = 0$, then the maximum is attained if $v_0 = +/-1$, which confirms that translational motion in a circle, of an arbitrary radius, allows equal adaptation for all γ locations at the conclusion of each period.

One additional possibility that assures $\int_0^{2\pi} \acute{\lambda}(\gamma, t)dt = 1$ for all values of γ is that of a bow-tie shaped periodic flight pattern, shown in figure 7.3.

Given again that $u(t) = \sin(t)$, but assuming $v(t) = v_0\cos(2t)$ the optimum value for v_0 is again +/- 1.

The first case scenario is the best to consider for actually moving the vehicle around. For a desired translational velocity, u_0, the persistence of excitation of the adaptive observer is sufficiently ensured by adding a sinusoidal velocity component, of any desired frequency, with an amplitude equal to $\sqrt{2}u_0$.

7.4.5 Analytical Results for Flight within a Tunnel Environment

In this section, the solution to the above adaptive observer assuming a tunnel-environment and a bow-tie flight pattern is evaluated. First, consider the adaptation law with no modeling the changing nearness and explicitly noting the dependence on time:

$$\dot{\widehat{\mu}}(\gamma,t) = -\rho\widetilde{\xi}(\gamma,t)\widehat{\lambda}(\gamma,t). \tag{7.29}$$

Using equation (7.11) as a substitution gives

$$\dot{\widehat{\mu}}(\gamma,t) = -\rho\widehat{\lambda}(\gamma,t)^2\widetilde{\mu}(\gamma,t) = -\rho\widehat{\lambda}(\gamma,t)^2\widehat{\mu}(\gamma,t) + \rho\widehat{\lambda}(\gamma,t)^2\mu(\gamma,t). \tag{7.30}$$

The right hand side in a form that is recognizable as the time varying differential equation is

$$\dot{\widehat{\mu}}(\gamma,t) + \rho\lambda(\gamma,t)^2\widehat{\mu}(\gamma) = \rho\lambda(\gamma,t)^2\mu(\gamma,t), \tag{7.31}$$

which has the solution

$$\hat{\mu}(\gamma,t) = \frac{\int p(\gamma,t)\rho\lambda(\gamma,t)^2\mu(\gamma,t)dt + \mu(\gamma,0)}{p(t)} \tag{7.32}$$

with

$$p(t) = e^{\int \rho\lambda(\gamma,t)^2 dt} \tag{7.33}$$

For an observer in tunnel environment, and assuming a constant orientation

in line with the inertial axes and with the origin located in the center of the tunnel, $\mu(\gamma)$ can be analytically defined as

$$\mu(\gamma, x, y) = \begin{cases} \frac{\sin(\gamma)}{a-y} & 0...\pi \\ \frac{-\sin\gamma}{a+y} & \pi...2\pi \end{cases} \tag{7.34}$$

For a bow-tie flight pattern, it is assumed that $\dot{x} = \sin(t)$ and $\dot{y} = \cos(2t)$ and hence the positions can be found by integrating the velocities, giving a position time history of

$$x(t) = -\cos(t) \tag{7.35}$$

$$y(t) = \sin(2t) \tag{7.36}$$

Using this, $\mu(\gamma, t)$, required for equation (7.32), can be found by substitution

$$\mu(\gamma, t) = \begin{cases} \frac{\sin(\gamma)}{a-\sin(2t)} & 0...\pi \\ \frac{-\sin\gamma}{a+\sin(2t)} & \pi...2\pi \end{cases} \tag{7.37}$$

The solution path $\widehat{\mu}(\gamma, t)$ in equation (7.32) is plotted from two sets of initial conditions, $\mu(\gamma, 0) = 1$ in figure 7.4 and $\mu(\gamma, 0) = 0$ in figure 7.5. In both cases, $a = 5$, and $\rho = 1$. Within 8 seconds at these initial conditions and the selected value of ρ, the characteristic double hump pattern associated with the nearness function of a tunnel is apparent.

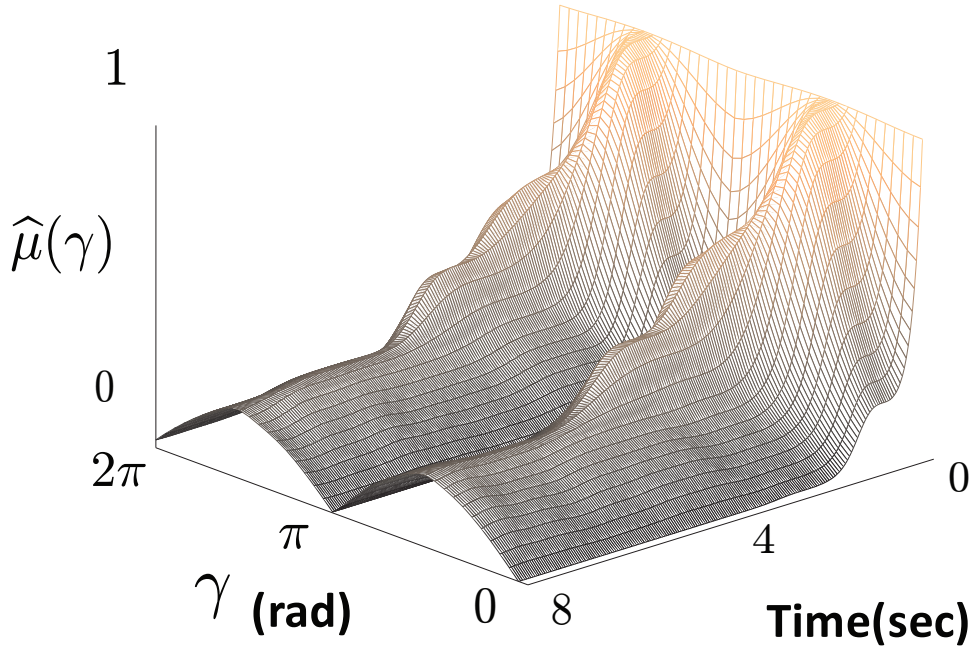

Figure 7.4: Solution for $\widehat{\mu}(\gamma, t)$ given $\widehat{\mu}(\gamma, 0) = 1$.

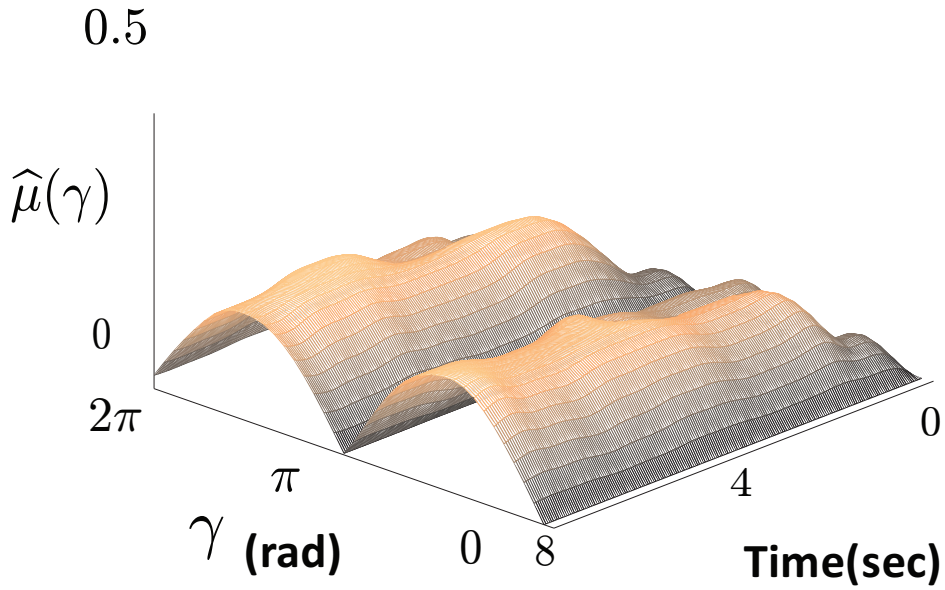

Figure 7.5: Solution for $\widehat{\mu}(\gamma, t)$ given $\widehat{\mu}(\gamma, 0) = 0$.

7.4.6 Discretization for Implementation

The continuous form of the observer, while having desirable properties theoretically, will typically require discretization in both space and time for practical implementation. The continuous nature of the theory allows the discretization to be performed at the last possible stage in the hardware implementation, thus supporting the continuous nature of computations performed using VLSI circuitry.

The spatially global persistency of excitation argument in section 7.4.3 utilized a discretized form of the nearness function, $\widehat{\mu}(\gamma)$ to permit an intuitive proof of exponential convergence. In most cases, it can be assumed that the 1D azimuthal optic flow function, $\xi(\gamma)$, will be available as spatially and temporally discrete, equidistantly sampled measurements. In this case it is assumed that

$$\xi[\gamma_k, n+1] = \int_{n\Delta_t}^{(n+1)\Delta_t} \int_{2\pi k/K}^{2\pi(k+1)/K} \frac{1}{\Delta_t} \xi(\gamma, t) d\gamma dt \tag{7.38}$$

which leads to a discrete implementation of the observer as

$$\widehat{\mu}[\gamma_k, n+1] = -\rho \widetilde{\xi}[\gamma_k, n] \widehat{\lambda}[\gamma_n, n] \Delta_t + \widehat{\mu}[\gamma_k, n] \tag{7.39}$$

7.4.7 Proof Assuming Presence of Structured Uncertainty

In the previous section, the presence of structures uncertainty and noise was neglected. Here, these disturbance sources are considered. As a preemptive method to ensure stability of the observer in the presence of noise and structured uncertainty, a *sigma modification* term is included in the adaptation law.

The adaptation law becomes

$$\dot{\widehat{\mu}}(\gamma) = \dot{\mu}_m(\gamma) - \rho\widetilde{\xi}(\gamma)\lambda - \rho\sigma\widehat{\mu}(\gamma). \tag{7.40}$$

Substituting this into (7.5),

$$\dot{W} = \frac{1}{\rho}\int_0^{2\pi} \widetilde{\mu}(\gamma)(\dot{\mu}_m(\gamma) - \dot{\mu}(\gamma)) - \rho\widetilde{\mu}(\gamma)\widetilde{\xi}(\gamma)\widehat{\lambda}(\gamma) - \rho\sigma\widetilde{\mu}(\gamma)\widehat{\mu}(\gamma)d\gamma \tag{7.41}$$

The middle term in the integrand contains $\widetilde{\mu}(\gamma)\lambda$ for which the substitution is

$$\widetilde{\xi}(\gamma) = \widetilde{\omega} + \widetilde{\mu}(\gamma)\widehat{\lambda}(\gamma) + \mu(\gamma)\widetilde{\lambda}(\gamma). \tag{7.42}$$

which includes the structural uncertainty induced by limited knowledge of the self-motion.

Using this and distributing $1/\rho$, $\dot{W}$ becomes:

$$\dot{W} = \int_0^{2\pi} -\widetilde{\mu}(\gamma)^2\widehat{\lambda}(\gamma)^2 - (\widetilde{\omega} + \mu(\gamma)\widetilde{\lambda}(\gamma))\widetilde{\mu}(\gamma)\widehat{\lambda}(\gamma) + \frac{(\dot{\mu}_m(\gamma) - \dot{\mu}(\gamma))\widetilde{\mu}(\gamma)}{\rho} - \sigma\widetilde{\mu}(\gamma)\widehat{\mu}(\gamma)d\gamma \tag{7.43}$$

After completing the square, the above becomes

$$\dot{W} \leq \int_0^{2\pi} -\frac{1}{2}\widetilde{\mu}(\gamma)^2\widehat{\lambda}(\gamma)^2 + \frac{1}{2}(\Delta_\omega + \Delta_{\lambda\mu})^2 - \frac{1}{2}\sigma\widetilde{\mu}(\gamma)^2 + \frac{1}{2}(\frac{\Delta_{\dot{\mu}}}{\rho} + \mu(\gamma)\sigma)^2 \tag{7.44}$$

Here, $\Delta_{\dot{\mu}}$, Δ_ω, and $\Delta_{\lambda\mu}$ denote the maximum expected absolute value of the terms $\dot{\mu}_m(\gamma) - \dot{\mu}(\gamma), \widetilde{\omega}$ and $\widetilde{\lambda}(\gamma)\mu(\gamma)$, respectively, over all values of γ.

If the persistency of excitation, requiring a non-zero $\widehat{\lambda}(\gamma)$, cannot be concluded to ensure exponential stability, the sigma modification term can do so, however it destroys the steady state convergence of the observer. In a practical application, the observer should be run first without the sigma modification and then included only if necessary.

7.5 Adaptation Simulation Results

7.5.1 Simulation of Entrance Identification

Given that insects utilize active vision to identify nest holes, one apt measure of the qualitative performance of the observer would be to use it to identify a similar type of entrance. For this simulation a vehicle is located in front of a tunnel-like entrance. In figures 7.8 and 7.8, the vehicle simply strafes back and forth perpendicular to the entrance. Note that convergence does not occur at the viewing angles located at the focii of expansion and contraction. In figures 7.9 and 7.10, the motion is the suggested prototypical bow-tie pattern, thus ensuring convergence at all viewing angles.

7.5.2 Simulation in a Tunnel Environment

A simulation was performed where an initialized nearness estimate was allowed to converge to the actual nearness over time. The simulation allowed the vehicle's trajectory to be specified and computed Lucas-Kanade optic flow as computed using OpenCV C functions on the imagery. The continuous-time algorithm was discretized

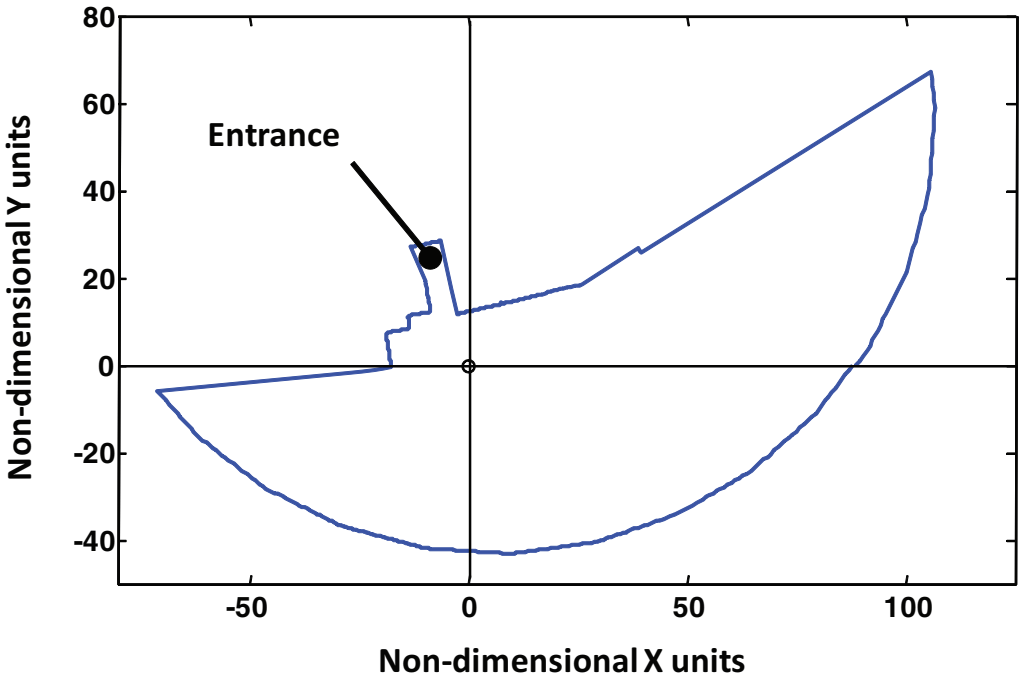

Figure 7.6: Top down view of Entrance Environment

and propagated in time using standard Euler integration.

Figure 7.11 shows the setup. The vehicle is positioned in the center of a tunnel-like environment. The vehicle executed a bow-tie pattern flight path reminiscent of insect zig-zagging behavior. With the estimated nearness initialized to be $\widehat{\mu}(\gamma) = 1$, after 80 iterations of the algorithm, figure 7.12 shows the resulting comparison between the estimated nearness function and the actual truth nearness. The value of the Lyapunov function, equation 7.5, is computed and plotted as a function of iteration number in figure 7.13. The norm of the estimate error is driven exponentially to zero. This simulation does not attempt to include the effects of $\dot{\mu}_m(\gamma)$ and therefore this neglected term will manifest itself as structured uncertainty and hence a disturbance to W.

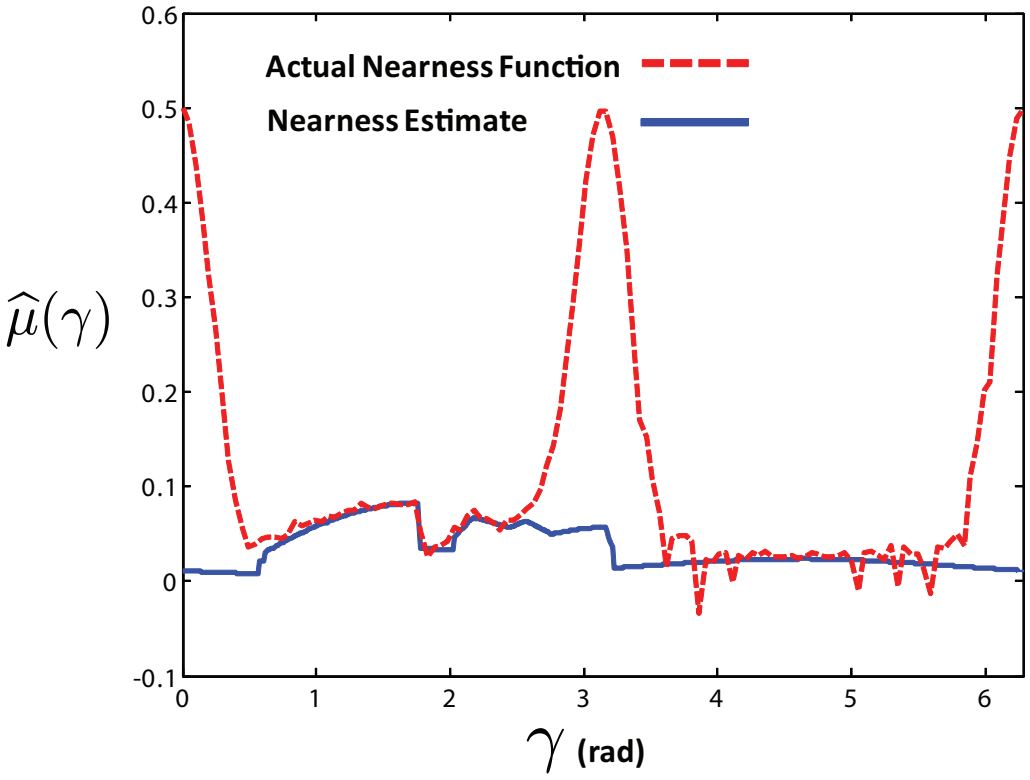

Figure 7.7: Converged Nearness Estimate Assuming 1-dimensional Motion

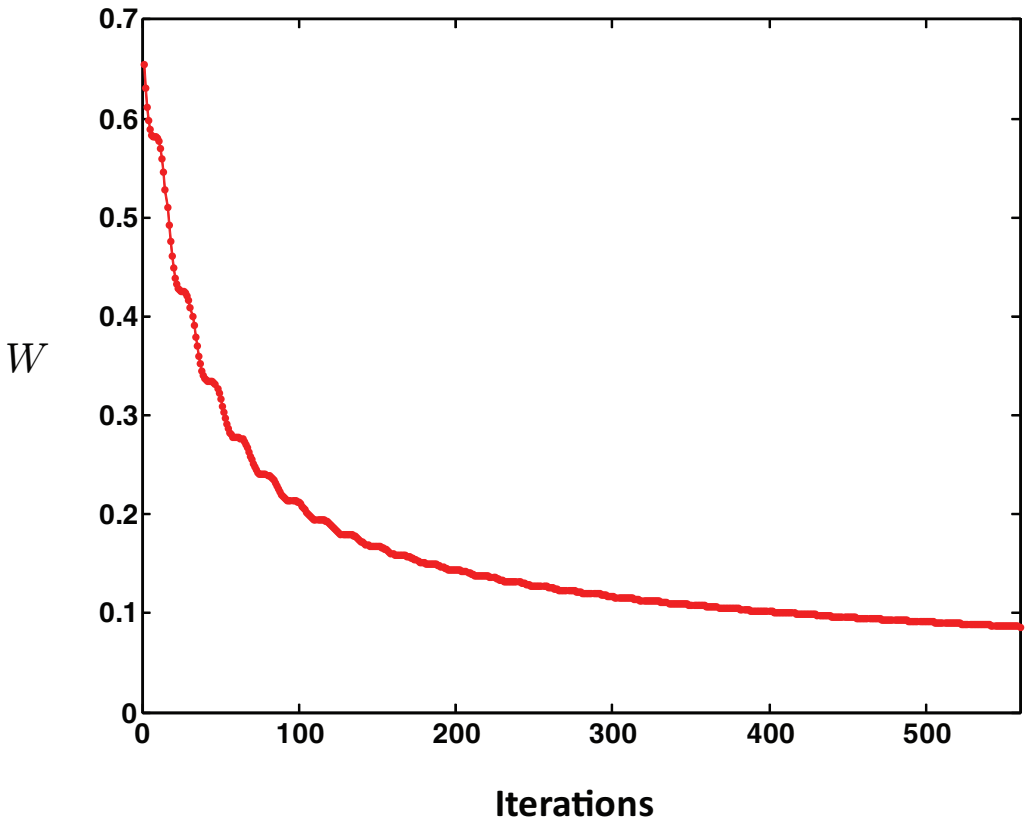

Figure 7.8: Lyapunov Energy as a Function of Iterations for 1D Motion

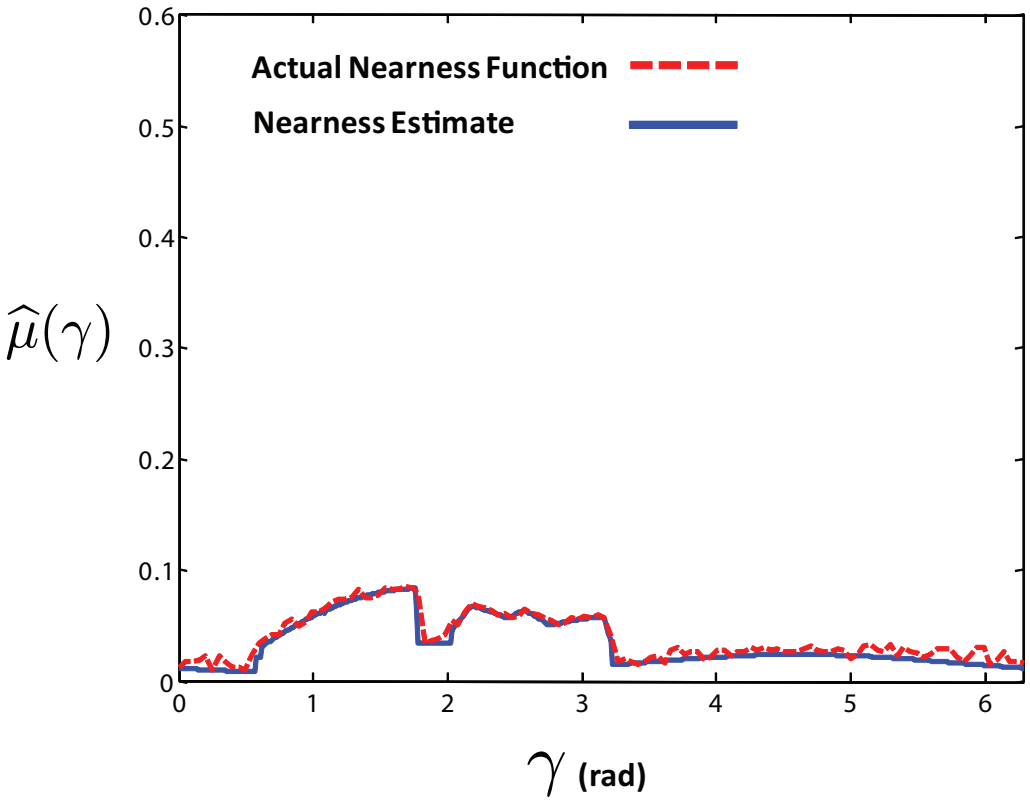

Figure 7.9: Converged Estimate Assuming Bow-tie Pattern

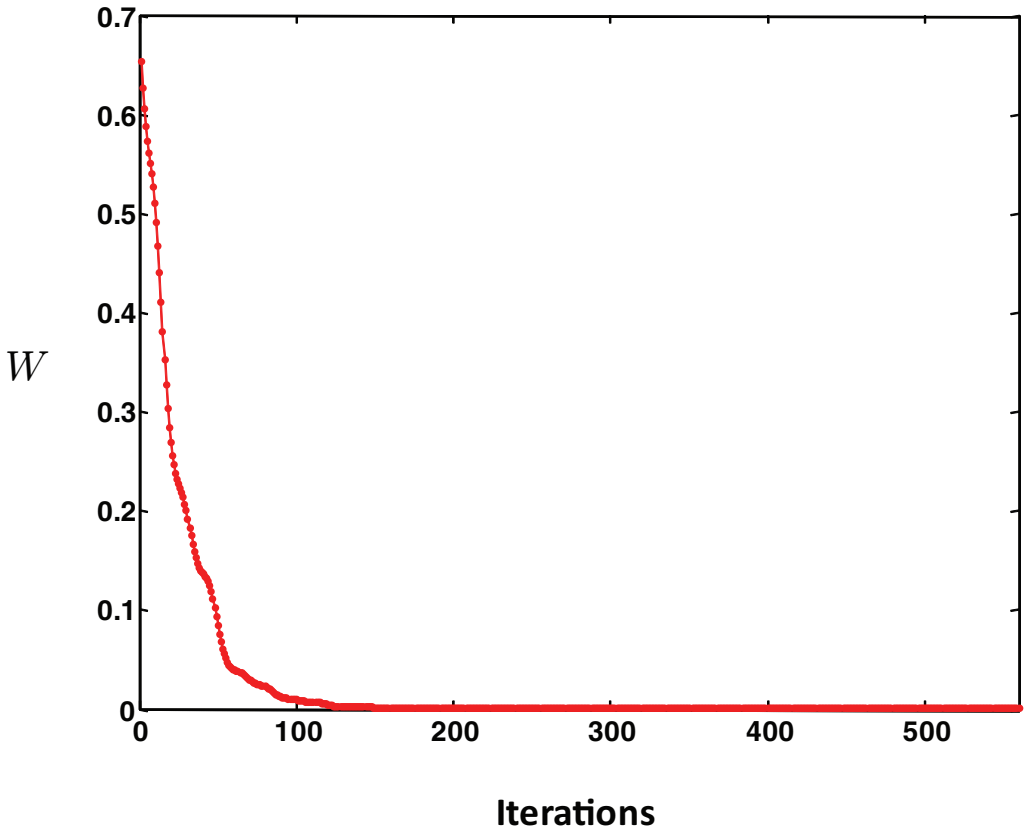

Figure 7.10: Lyapunov Energy as a Function of Iterations for Bow-tie Flight Pattern

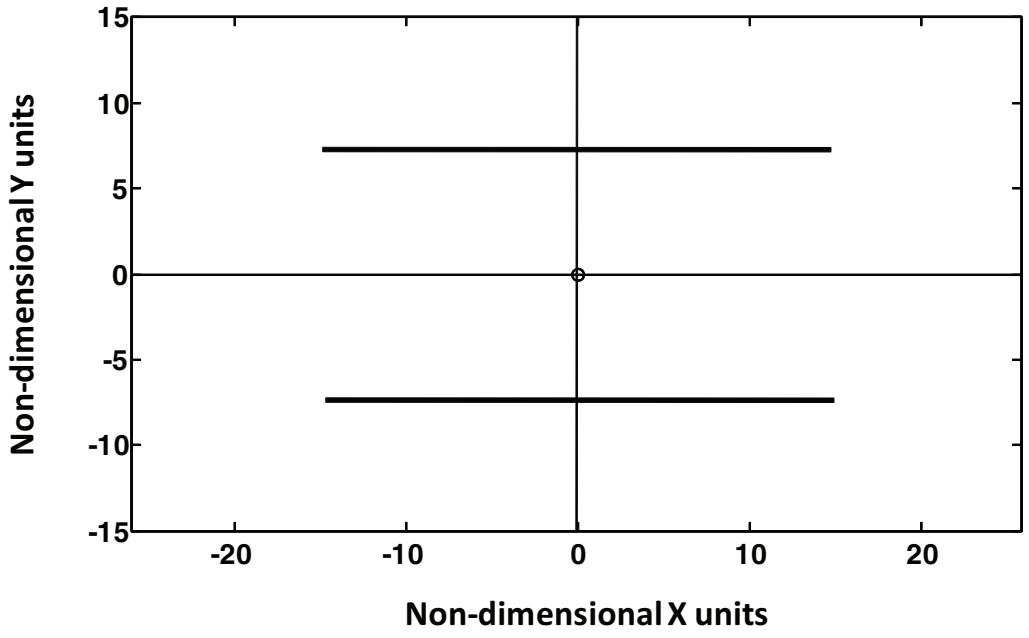

Figure 7.11: Top down view of Tunnel Environment

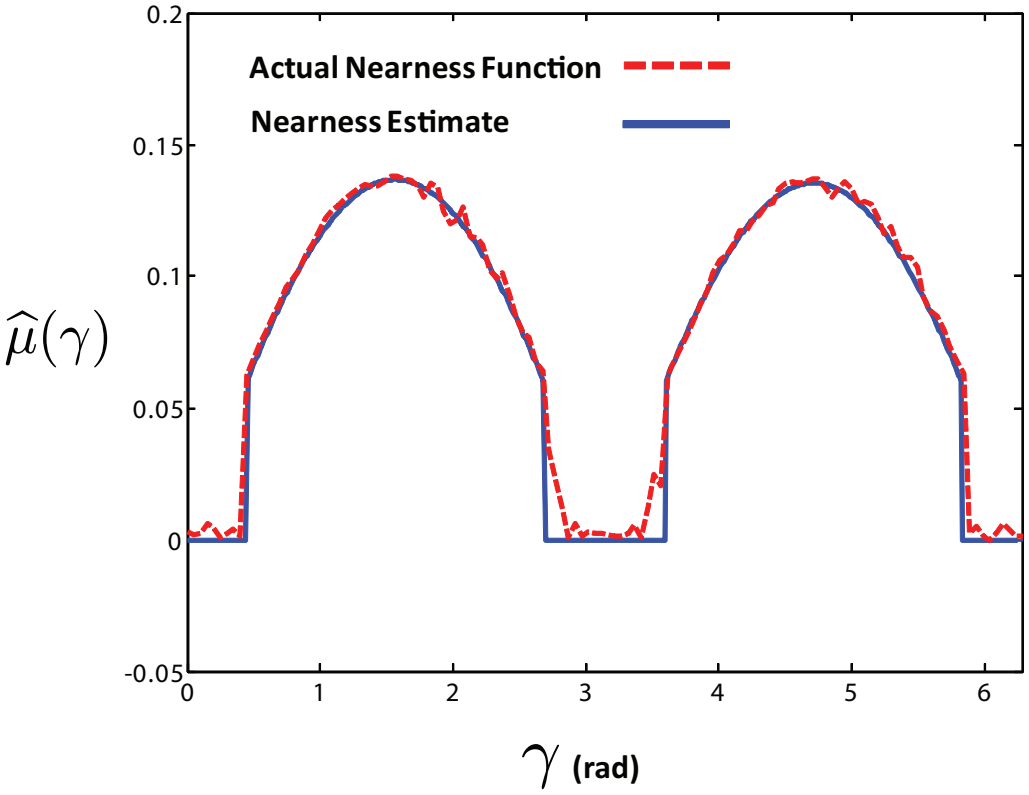

Figure 7.12: Converged Nearness Estimate Assuming Bow-tie Pattern

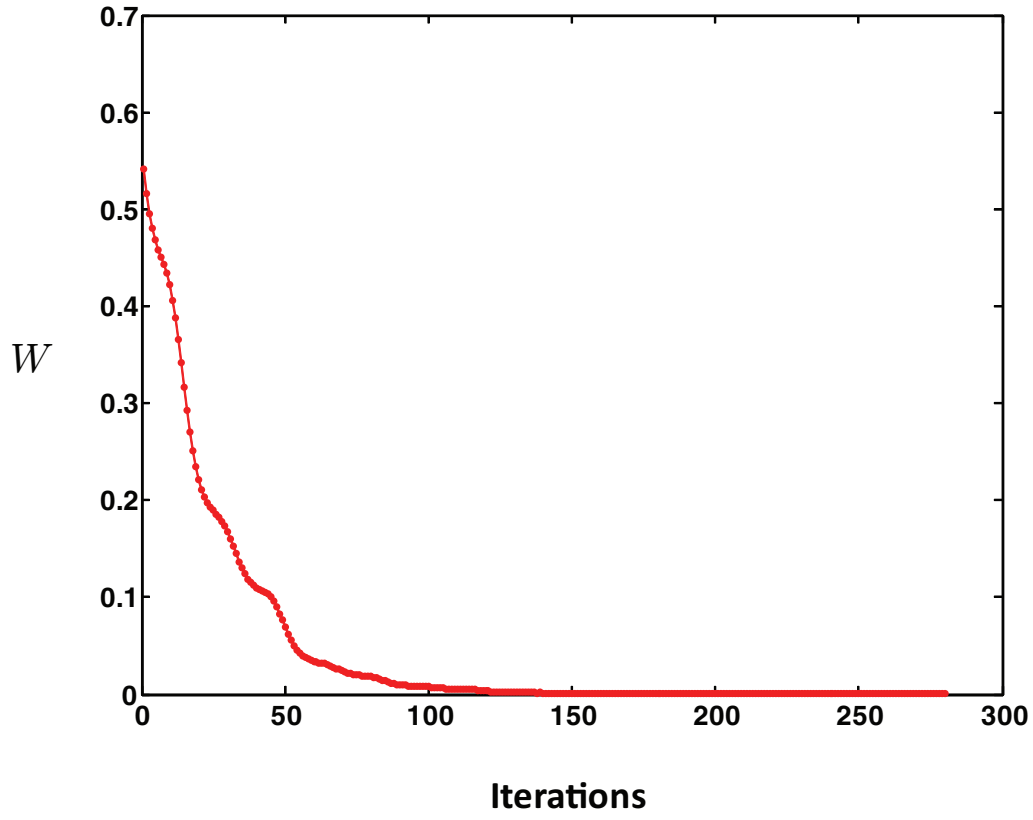

Figure 7.13: Lyapunov Energy as a Function of Iterations Assuming Bow-tie Flight Pattern

7.5.3 Simulation within Bent Corridor

The prototypical bow-tie motion was repeated for the vehicle in a corridor environment with a 90 degree bend. Figure 7.14 shows a top down view of the environment. Figure 7.15 shows the resulting converged nearness function compared with the actual truth nearness. Finally, the Lyapunov function is plotted as a function of time in figure 7.16.

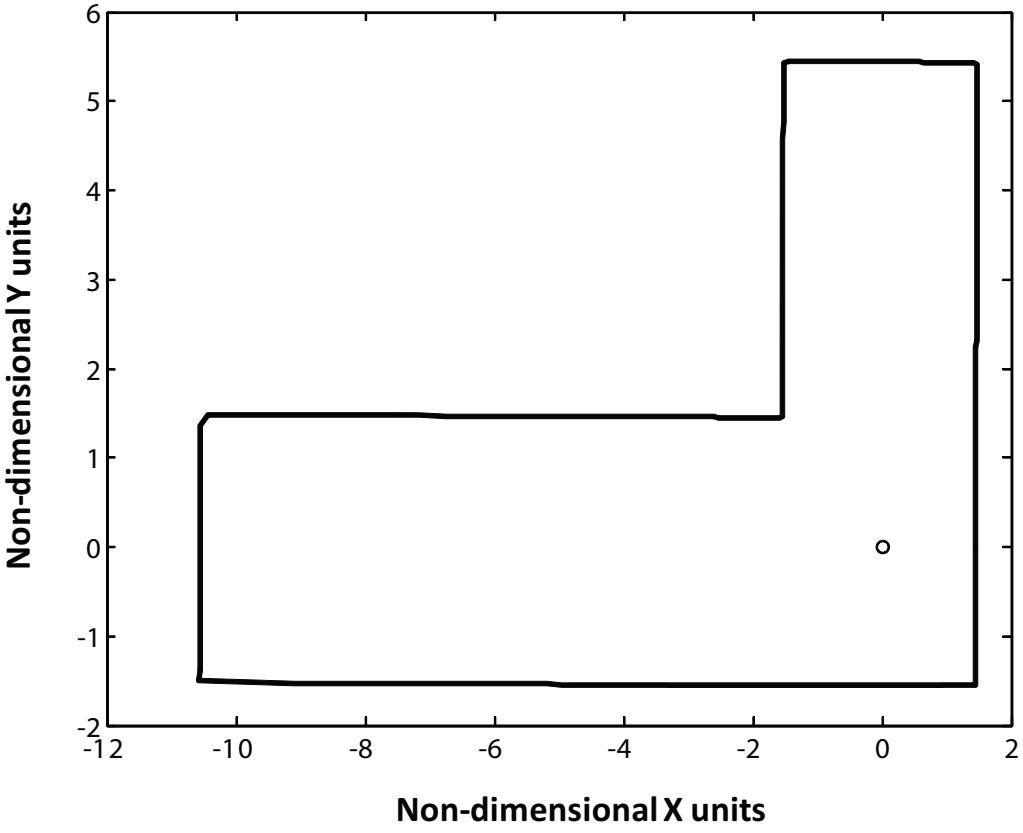

Figure 7.14: Top Down View of Bent Corridor Environment

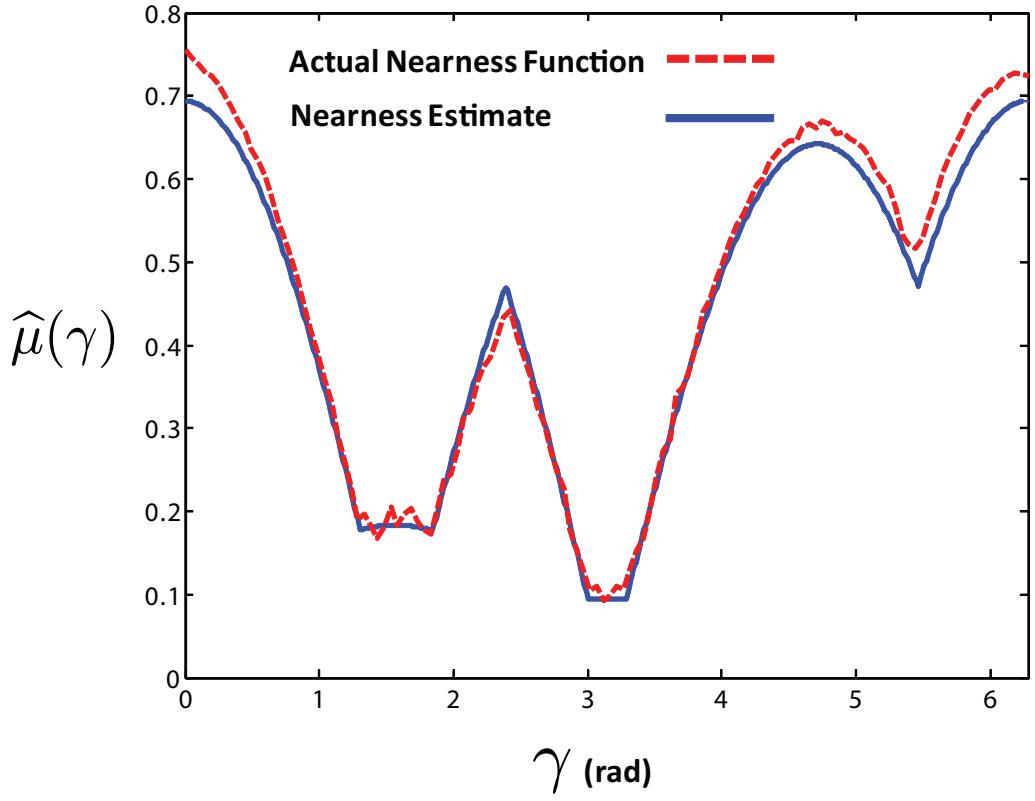

Figure 7.15: Converged Nearness Estimate Assuming Bow-tie Pattern

7.5.4 Simulation within a Forest Environment

To test the ability of the observer to detect small obstacles, the simulation was run similar to the previous runs, however using a "forest-like" environment comprised of various pillars. The top down plotted actual nearness is shown in figure 7.17. The converged estimate is shown in figure 7.18, and the plotted Lyapunov function over time is shown in figure 7.19.

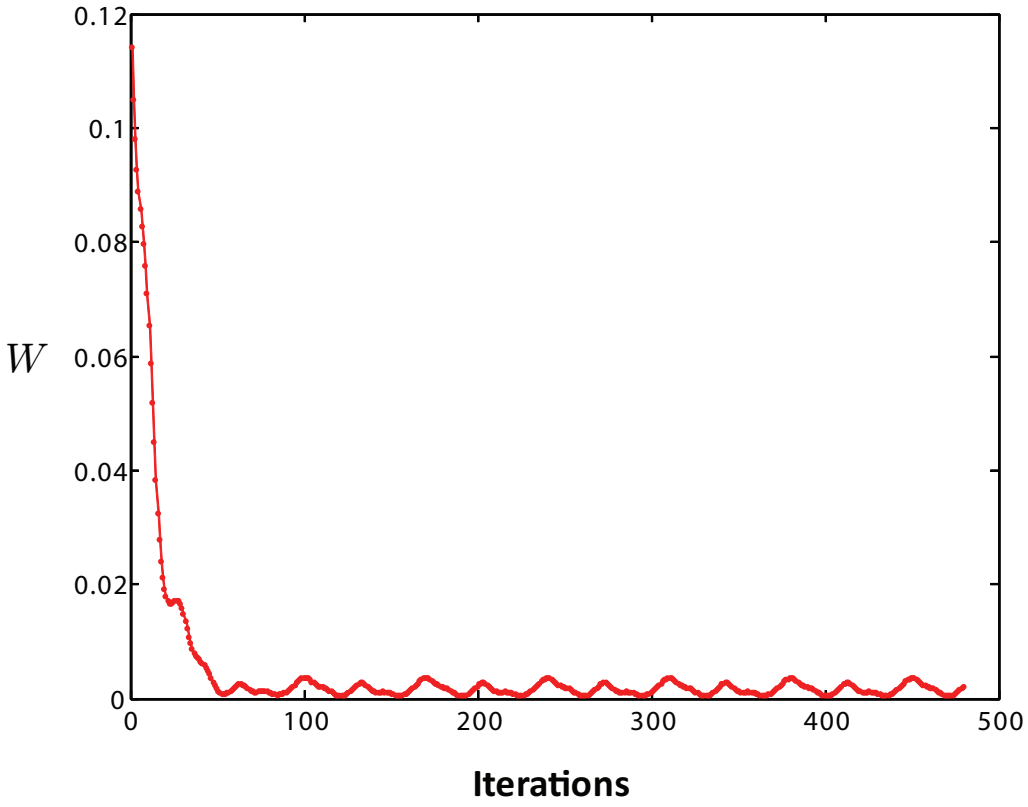

Figure 7.16: Lyapunov Energy as a Function of Iterations Assuming Bow-tie Flight Pattern

7.5.5 Navigation using the Cost Function Minimization Method

Given a particular distribution of obstacles surrounding a vehicle, represented by a nearness function $\mu(\gamma)$, one possible direction of motion would be away from the closest obstacles and towards open areas, excluding any higher level guidance requirements. The optimal solution for this direction can be determined by minimizing a cost function that considers obstacle location and nearness. The cost of heading in a particular direction, γ_d, is determined by considering how close an obstacle is and the difference in angular orientation between its location on the retina

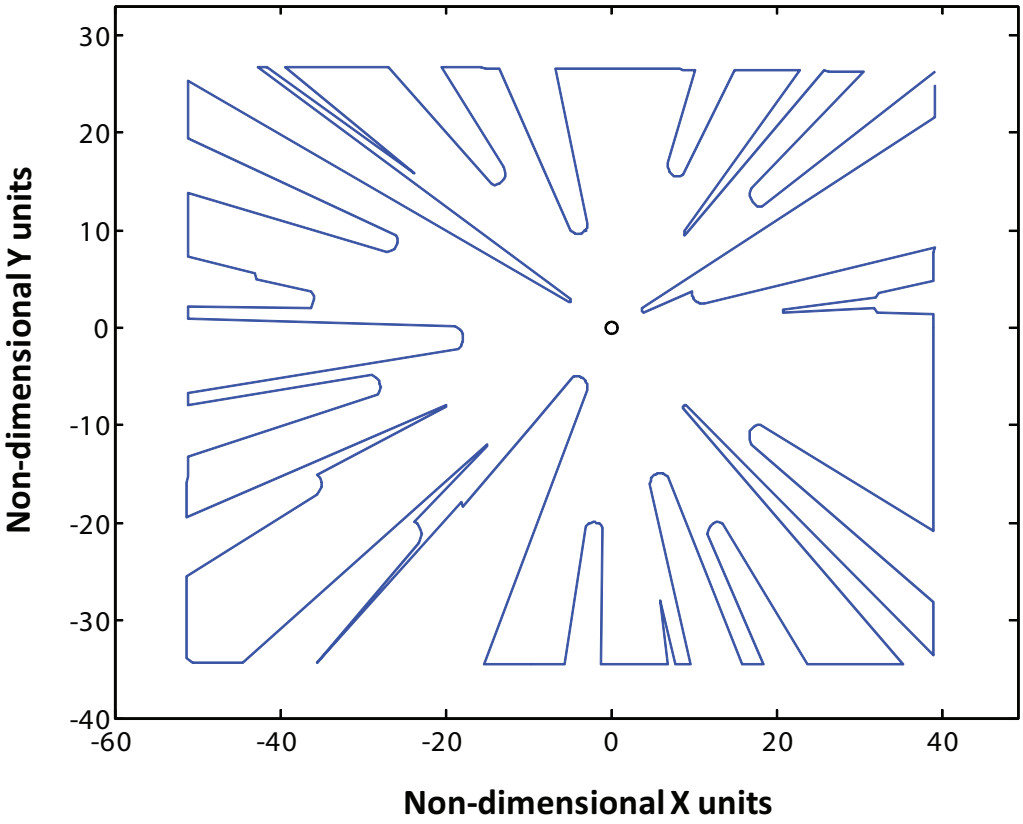

Figure 7.17: Top Down View of "Forest" Environment

and γ_d. The cost is accumulated by considering all obstacles on the retina at once. A suitable cost function can be constructed as

$$S(\gamma_d) = \int_0^{2\pi} \mu(\gamma) e^{-a_0(\gamma_d - \gamma)^2}. \tag{7.45}$$

The constant a_0 in this is a decay parameter that sets the sensitivity of the cost to angular distance between γ_d and an object at γ.

It can be recognized that equation 7.45 can be rewritten as the convolution of $\mu(\gamma)$ with a Gaussian kernel:

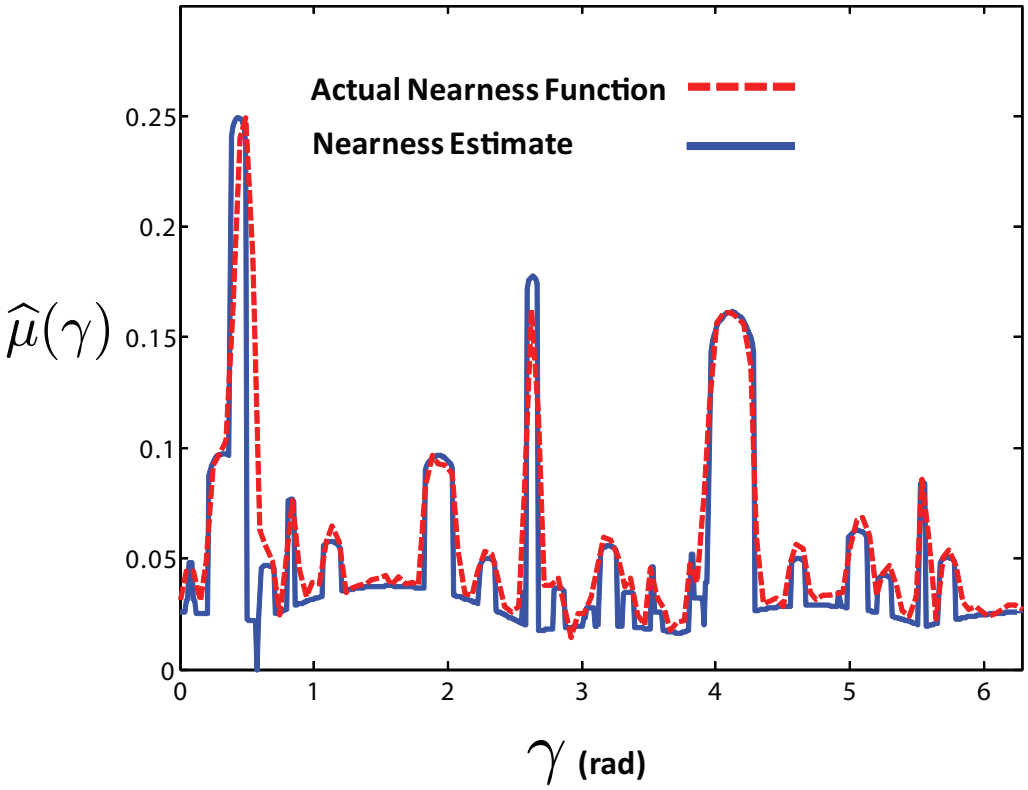

Figure 7.18: Converged Nearness Estimate Assuming Bow-tie Pattern

$$S(\gamma_d) = \mu(\gamma) \star e^{-a_0(\gamma_d - \gamma)^2}. \tag{7.46}$$

The optimal direction would involve minimizing this cost function. A gradient descent formulation for finding the minimum allows the creation of a differential equation for the estimation of the minimum. This can be given as

$$\dot{\widehat{\gamma}}_{min} = -b_0 \frac{\partial S(\gamma_d)}{\partial \gamma_d} \tag{7.47}$$

with b_0 as a positive constant, and $\widehat{\gamma}_{min} \in [0, 2\pi]$.

Unless there is only one minimum for $S(\gamma_d)$, the minimization strategy of equation 7.47 will lead only to a local minimum. A local minimum is typically an

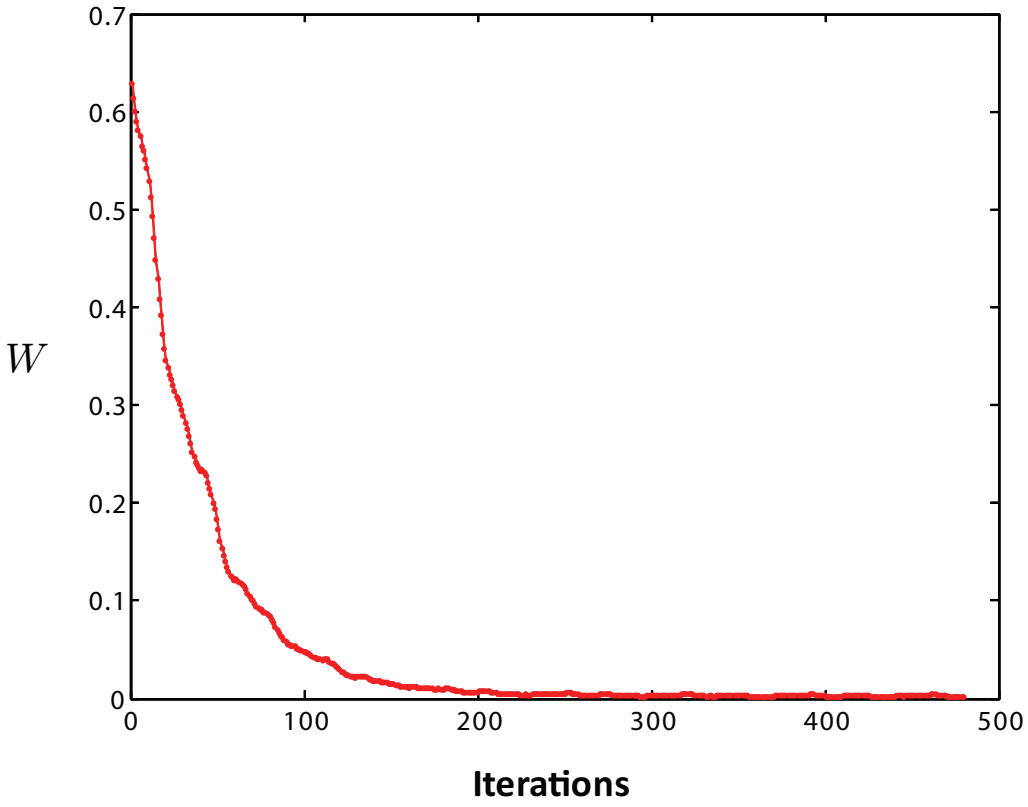

Figure 7.19: Lyapunov Energy as a Function of Iterations Assuming Bow-tie Flight Pattern

acceptable direction of travel, so long as the magnitude of $S(\gamma_d)$ at that minimum is below a certain acceptable level. To prevent a collision brought on by a local minimum that is too high relative to other possible directions, a "saccade" reaction can be created. If the local minimum grows beyond a certain threshold, the *global* minimum of $S(\gamma_d)$ is computed and $\widehat{\gamma}_{min}$ is set to this location, thus initiating a sharp "step" change in the desired direction of travel.

To address the possibility that the minimum seeking real-time optimization algorithm is stuck in a local minimum that represents an impending collision, a saccade reaction is included that causes the vehicle to switch its desired direction of

travel to the global minimum via a brute force search. This is a necessary inclusion to compliment the continuous minimum seeking nature of the minimum seeking algorithm.

7.5.5.1 Navigation within a Forest Environment

The above developed basic navigation algorithm was simulated in a forest-like environment. The simulation results are shown in figure 7.20. Note that the vehicle doubles back on its initial path. A primary purpose of this simulation is to demonstrate that while the observer does not explicitly account for the continuous change in the actual nearness function, the observer is still capable of performing well in a dynamically changing environment so long as the gain ρ is sufficiently high.

7.6 Experimental Validation on a Flying Quadrotor Platform

This section presents the validation of the observer using an actual flying testbed platform. The azimuthal optic flow was provided by the Centeye Faraya Ring, providing 56 optic flow measurements, shown in figure 7.21. The angular spacing between optic flow elements was assumed to be constant, although slight offsets due to manufacturing tolerances were present. The individual sensor fields of view overlap slightly, ensuring complete coverage of the azimuth.

The raw output of the sensors needed to be calibrated to physical units. This calibration was performed using a yaw rate table test stand, shown in figure 7.22. A representative optic flow output curve is shown in figure 7.23. A linear calibration

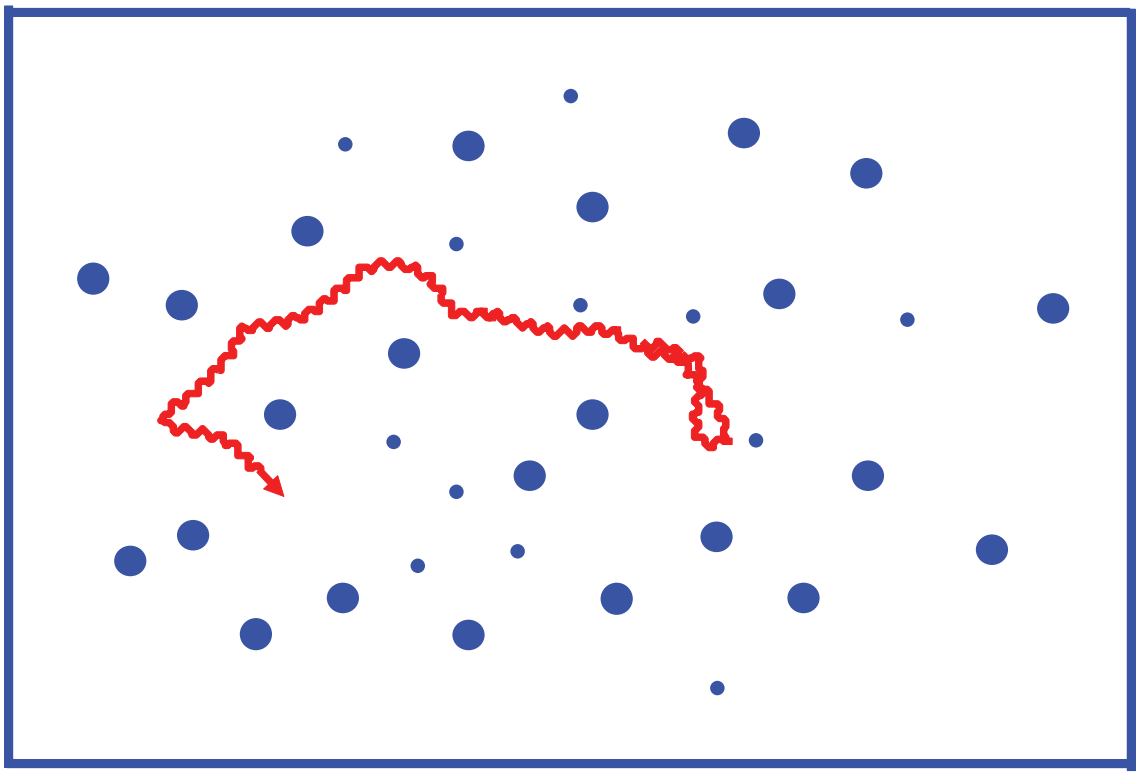

Figure 7.20: Simulation of Navigation in a Forest-like Environment

curve was estimated for each sensor independently using the approximately linear range of $[-2, 2]$ rad/s.

A top down view of the experimental setup is shown in figure 7.24. A diagram of this setup is shown in 7.25. The objective was to identify the azimuthal angular location and approximate nearness of the three pillar obstacles using small deviations from the starting position.

The estimated nearness after approximately 23 seconds of flight is shown in 7.26. The influence of the three pillars can be seen, however a slight offset between the actual nearness and the estimated nearness is noted. This is due to the perceived azimuthal locations of the pillars changing as the vehicle moves. The resulting estimate reflects a spatial averaging effect. This can be reduced if the obstacles are at a greater distance given the same pattern of self-motion, or if the pattern of self-motion deviates less from the starting position. Furthermore, a distinct over prediction in the nearness is noted. An overprediction in the nearness is due to a over prediction in the optic flow. The likely reason for this is an over prediction in the linear calibration curve within the range of operation that the sensors were used for the test. Finally, while the indicated "actual" nearness function is zero between the pillar obstacles for illustrative reasons, it is apparent from the actual photo, figure 7.24, that the walls and boxes around the edge of the room are close enough to account for the non-zero nearness between obstacle azimuthal locations in 7.26. Nevertheless, this experimental implementation, as well as the simulations, demonstrate that the structure from motion observer presented in this thesis shows promise as a realistic approach to navigation.

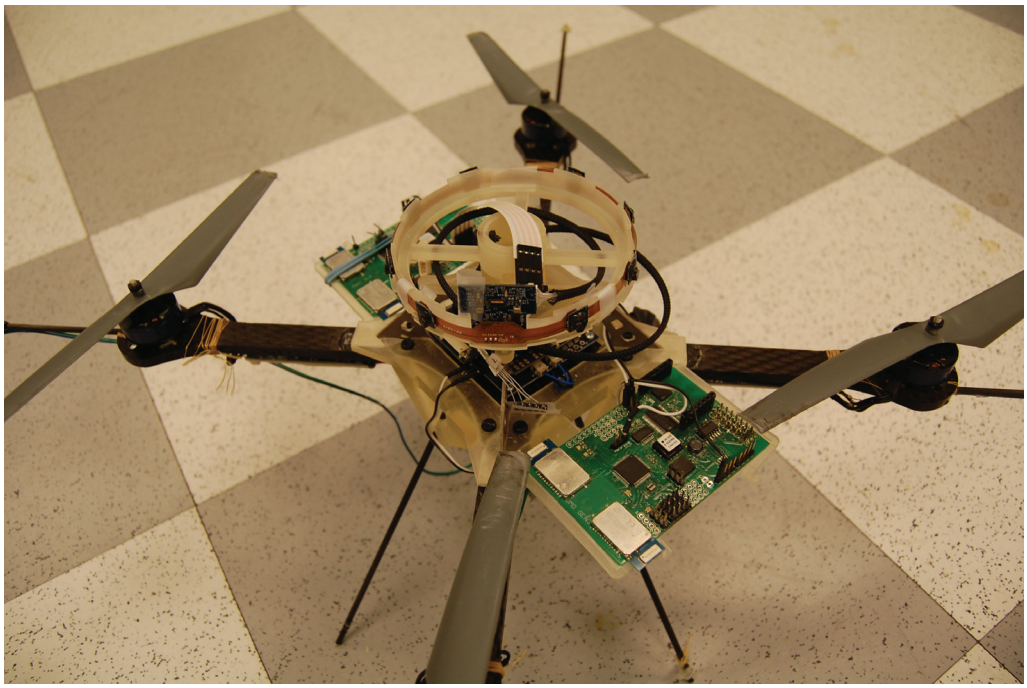

Figure 7.21: Quadrotor Experimental Setup with Centeye Faraya Sensor Ring

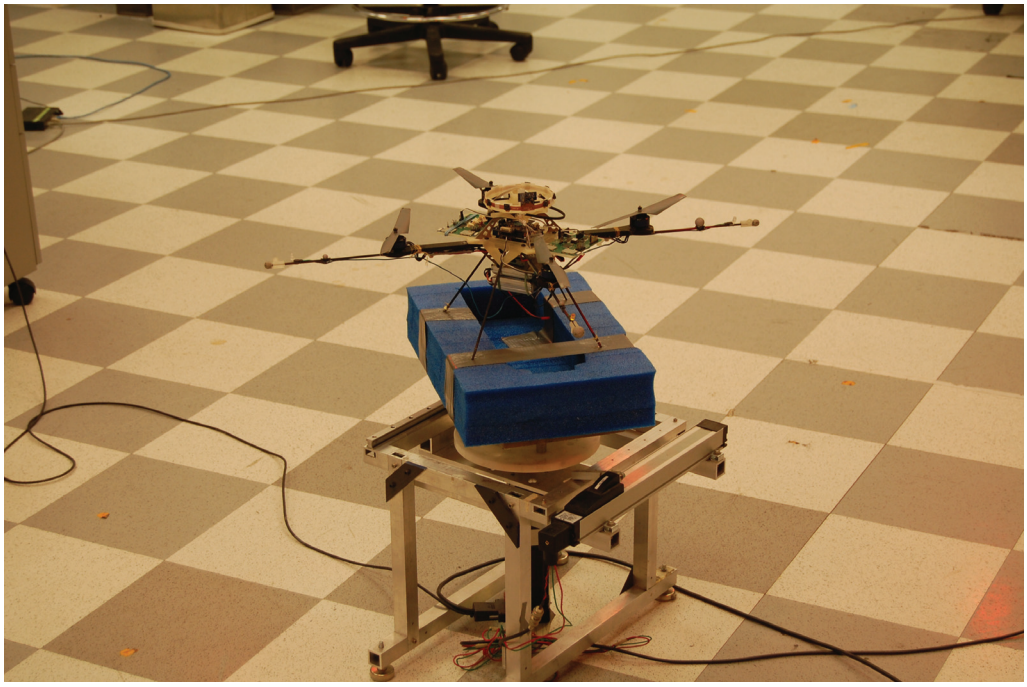

Figure 7.22: Calibration Yaw Rate Test Stand

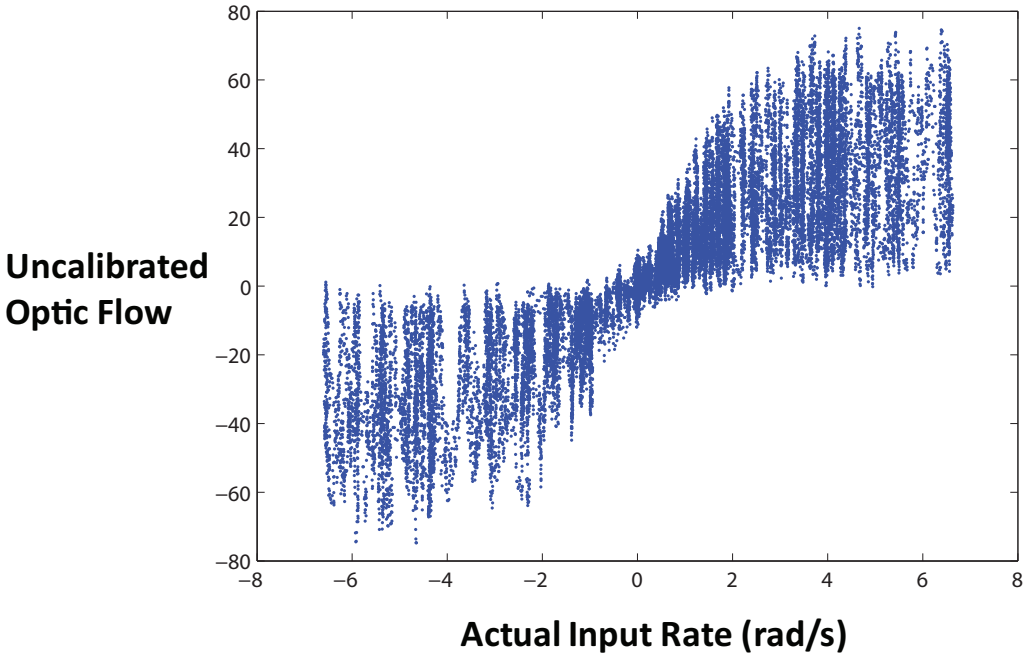

Figure 7.23: Example Sensor Calibration

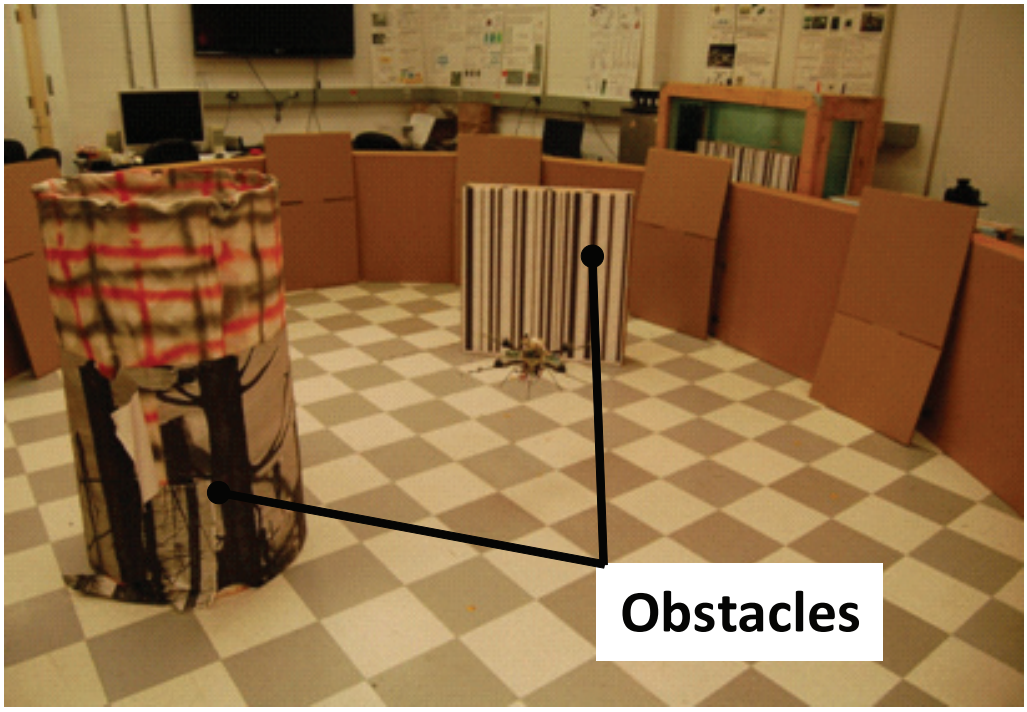

Figure 7.24: Quadrotor Experimental Setup Top Down Photo

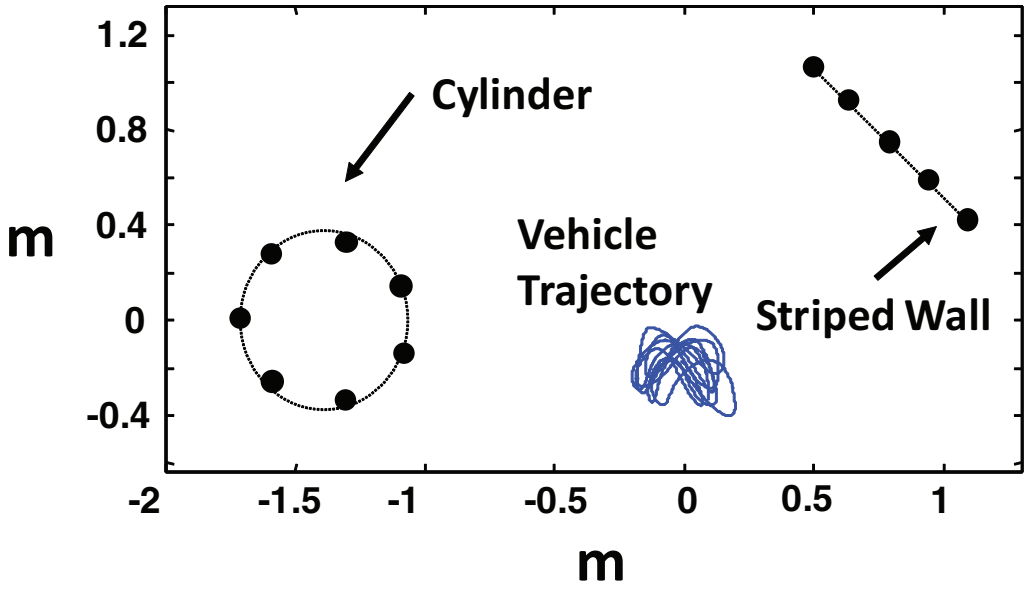

Figure 7.25: Experimental Flight Top Down View Diagram

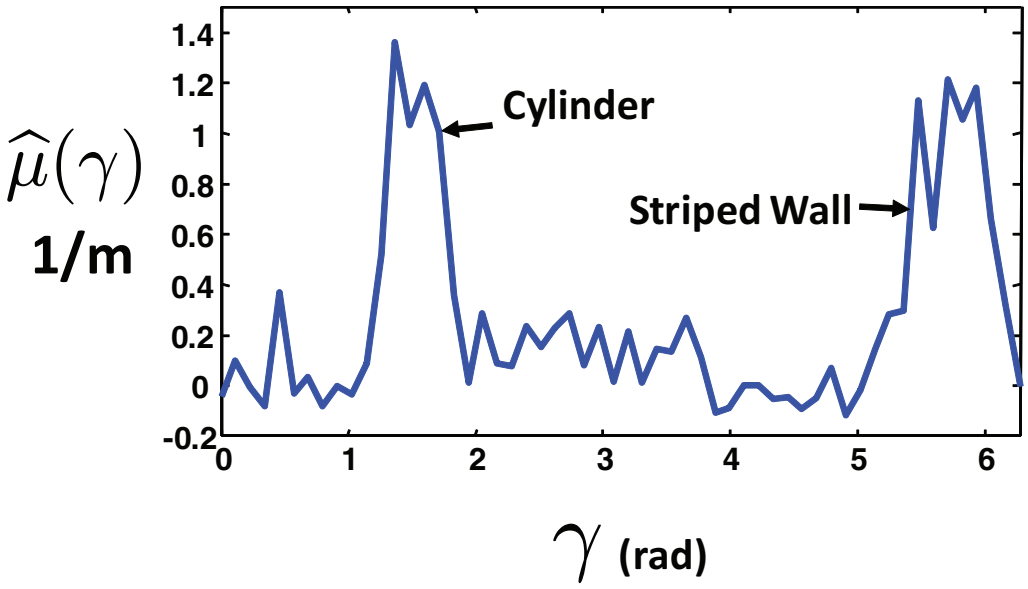

Figure 7.26: Converged Nearness for Experimental Flight

Chapter 8

Conclusions and Future Work

While the primary focus of this thesis involves visual navigation, it also addresses many aspects of a complete micro air vehicle control and navigation system. A motivating factor throughout this research was the desire to physically implement developed theory to demonstrate the feasibility. To accomplish this goal, several contributions involved developing and advancing the prerequisite theory and hardware comprising actual experimental MAV testbed experiments.

A primary prerequisite for attempting to implement visual navigation theory is the availability of a sufficiently stable platform with which to work. A stable platform requires that many subsystems work together efficiently. For this reason, this thesis also presents advances in avionics development, feedback control, and state estimation, as well as the demonstration of the practicality of using system identification to estimate useful yet minimum complexity flight dynamics models for MAVs.

8.1 State Sensing

Several generations of avionics systems were custom developed to facilitate the implementation of autonomy on the desired testbed vehicles. Unlike commercially available models, emphasis was placed on ease of integration and telemetry

bandwidth.

8.2 System Identification

System identification was performed on the Honeybee single main rotor electric helicopter. Investigation of the identified vehicle dynamics showed that the *hybrid model* typically used for large scale helicopters is also applicable to these micro rotorcraft. This identification also allowed the assumption of the cyclic, heave, and yaw dynamics as independent dynamic models for control design purposes.

8.3 Feedback Control

Feedback control was implemented both on-board and off-board as necessary to support the demonstration of the desired visual navigation experiments. The lateral/longitudinal dynamics, heave dynamics, and yaw dynamics were decoupled and treated as separate systems, allowing certain degrees of freedom to be controlled using off-board feedback and others to be controlled using on-board feedback as desired. Most typically, a particular desired height was maintained to permit the assumption of planar motion in the environment.

8.4 State Estimation

State estimation algorithms were used primarily to permit an attitude estimate using on-board inertial sensing. The developed visual algorithms, however, can be viewed as a form of state estimation as well, with proximity to obstacles as being

the "states".

8.4.1 Relative Proximity Navigation using Wide-Field Integration of Optic Flow

The visual navigation development was implemented as an outer loop to the inner loop control that stabilized the vehicle. Prior developed wide-field integration theory was demonstrated successfully for the first time on a 6 DOF flying vehicle. This successfully demonstrated the mimicry of the neural architecture and behavior of insects in an artificial flying robot.

8.4.2 Structure from Motion Observer

The core contribution of this thesis, however, is the theoretical development of a nonlinear observer for structure from motion with omni-directional spatial exponential convergence, on average, assuming the persistency of excitation condition is met. The proposed observer is ideal for vehicles that exhibit hovering and unconstrained motion through the environment due to the usage of omni-directional motion estimation. The persistency of excitation condition can be intuitively thought of as not allowing the focus of expansion and focus of contraction to remain at the same viewing angles for too long. In fact, best performance of the observer, in terms of omni-directional spatial convergence, is possible if the vehicle follows certain prototypical motions. The persistency of excitation requirement provides mathematical insight as to why some insect species perform zig-zagging flight pat-

terns to extract structure from the environment, through a process known as active vision. Structured uncertainty regarding the self-motion of the vehicle or how the environment changes with position, as well as random noise, will be manifest as disturbances to the Lyapunov function. These disturbances will not destroy the inherent exponential stability of the observer.

8.5 Future Work

This observer was implemented in simulation and experimentally and has been demonstrated to be a feasible, practical, and realistic approach to optic flow-based obstacle detection. This initial theoretical design can be improved upon in several ways. It may not be the case that any knowledge of the self-motion is available, either through direct measurement or through knowledge of a dynamic model. In this case, the observer must be modified to allow the optic flow to estimate the self-motion as well, which would result in well known structure and motion identification ambiguities. The observer can be extended to the case where other measurements, such as absolute distance sensors or radar, can be incorporated to aid in this case.

Furthermore, one obvious deficiency of the observer is the restriction to planar motion. To accommodate more aggressive maneuvering, the observer should be extended to the cylindrical, or more elegantly, the spherical case, utilizing 3D omni-directional optic flow.

Finally, while the observer recognizes that the actual nearness of the environment does change, according to $\dot{\mu}(\gamma)$, there is no explicit attempt in this thesis

to account for it. A modified approach would be to attempt to explicitly account for the change in the nearness as the vehicle's inertial position changes, through inclusion of inertial states in the observer.

Given the addition of a model of $\dot{\mu}(\gamma)$, the observer is ideally suited for the detection of small obstacles from optic flow. As a small object moves across the retina, or imaging space, if no attempt is made to capture where the object is expected to be on the retina as a function of time, the influence of the optic flow generated by that object tends to be averaged over the spatial domain along which it moved.

Bibliography

[1] Eric R. Bachmann, Robert B. McGhee, Xiaoping Yun, and Michael J. Zyda. Inertial and magnetic posture tracking for inserting humans into networked virtual environments. In *VRST '01: Proceedings of the ACM symposium on Virtual reality software and technology*, pages 9–16, New York, NY, USA, 2001. ACM.

[2] Anna Mulrine. Drones fill the troops gap in Afghanistan. U.S. News & World Report, September 2008.

[3] Roy Braybrook. Air ops. Armada International, December 2008.

[4] David Collogan. UAVs on the horizon. Business & Commercial Aviation, July 2006.

[5] Omead Amidi, Yuji Mesaki, and Takeo Kanade. A visual odometer for autonomous helicopter flight. *Robotics & Autonomous Systems*, 28:185, 1999.

[6] Matt Garratt and Javaan Chahl. Visual control of an autonomous helicopter. In *Proceedings of the 41st Aerospace Sciences MEeting and Exhibit*. Reno, Nevada, 2003.

[7] Stefan Hrabar and Gaurav S. Sukhatme. Omnidirectional vision for an autonomous helicopter. In *In IEEE International Conference on Robotics and Automation*, pages 558–563, 2003.

[8] Pedro J. Garcia-Pardo, Gaurav S. Sukhatme, and James F. Montgomery. Towards vision-based safe landing for an autonomous helicopter. *Robotics and Autonomous Systems*, 38(1):19 – 29, 2002.

[9] Jian Li and Rama Chellappa. A factorization method for structure from planar motion. *Motion and Video Computing, 2005. WACV/MOTIONS '05 Volume 2. IEEE Workshop on*, 2:154–159, January 2005.

[10] P Gurfil and H Rotstein. Partial aircraft state estimation from visual motion using the subspace constraints approach. *Journal of Guidance, Control, and Dynamics*, 24:1016–1028, 2001.

[11] F Kendoula, Isabelle Fantoni, and Kenzo Nonamib. Optic flow-based vision system for autonomous 3d localization and control of small aerial vehicles. *Robotics and Autonomous Systems (in press)*, 2009.

[12] Matthias O. Franz, Javaan S. Chahl, and Holger G. Krapp. Insect-inspired estimation of egomotion. *Neural Comput.*, 16(11):2245–2260, 2004.

[13] Juergen Haag and Alexander Borst. Electrical Coupling of Lobula Plate Tangential Cells to a Heterolateral Motion-Sensitive Neuron in the Fly. *J. Neurosci.*, 28(53):14435–14442, 2008.

[14] Sreeja Rejesh, David C. O'Carroll, and Derek Abbott. Man-made velocity estimators based on insect vision. *Journal of Smart Materials and Structures*, 14, 2005.

[15] H. Cuntz and A. Borst. Neural image processing by dentritic networks. In *Proceedings of the National Academy of Sciences*, volume 100. 2003.

[16] M. Egelhaaf and A. Borst. Motion computation and visual orientation in flies. *Comp. Biochem. Phisiol.*, 104A:659–673, 1993.

[17] R. M. Olberg, A. H. Worthington, and K. R. Venator. Prey pursuit and interception in dragonflies. *Journal of Comparative Physiology A: Neuroethology, Sensory, Neural, and Behavioral Physiology*, 186:155–162, 2000.

[18] P.V. Reddy, E.W. Justh, and P.S. Krishnaprasad. Motion camouflage in three dimensions. In *Decision and Control, 2006 45th IEEE Conference on*, pages 3327 –3332, 2006.

[19] Alexander Borst, Juergen Haag, and Dierk F. Reiff. Fly motion vision. *Annual Review of Neuroscience*, 33(1):49–70, 2010.

[20] J. Conroy and D. Pines. Development of a micro air vehicle avionics package for system identification and vehicle control applications. In *Proceedings of the American Helicopter Society Unmanned Rotorcraft Specialist's Meeting*, Chandler, AZ, Jan 23-25 2007.

[21] D. Gebre-egziabher, G. H. Elkaim, J. D. Powell, and B. W. Parkinson. A non-linear, two-step estimation algorithm for calibrating solid-state strapdown magnetometers. In *in 8th International St. Petersburg Conference on Navigation Systems*, pages 28–30, 2001.

[22] Demoz Gebre-Egziabher, Gabriel H. Elkaim, J. David Powell, and Bradford W. Parkinson. Calibration of strapdown magnetometers in magnetic field domain. *Journal of Aerospace Engineering*, 19(2):87–102, 2006.

[23] Jared Grauer, Joseph Conroy, J. Hubbard, J.S. Humbert, and D. Pines. Field calibration of inertial measurement units for miniature unmanned aircraft. In *AIAA Paper No. 2009-727*, 2009.

[24] Geoffrey L. Barrows, Javaan S. Chahl, and Yam V. Srinivasan. Biomimetic visual sensing and flight control. In *Proc. Bristol UAV Conf*, pages 159–168, 2002.

[25] William E. Green, Paul Y. Oh, and Geoffrey Barrows. Flying insect inspired vision for autonomous aerial robot. In *Maneuvers in Near-Earth Environments, IEEE International Conference of Robotics and Automation*, pages 2347–2352, 2004.

[26] Ajit Singh. *Optic Flow: A Unified Perspective*. IEEE Computer Society Press, Los Alamitos, CA, 1991.

[27] W. Reichardt. Evaluation of optical motion information by movement detectors. *Biol. Cybern.*, 56:247–254, 1997.

[28] Joseph Kearney, William Thompson, and Daniel Boley. Optical flow estimation: An error analysis of gradient-based methods with local optimization. *Pattern Analysis and Machine Intelligence*, 1987.

[29] B. Pamadi. *Performance, Stability, Dynamics, and Control of Airplanes*. American Institute of Aeronautics and Astronautics, 1998.

[30] J. B. Kuipers. *Quaternions and Rotation Sequences*. Princeton University Press, Princeton, New Jersey, 1999.

[31] G. Wahba. A least-squares estimate of satellite attitude. *SIAM Review*, 7(3):409, July 1965.

[32] C. Hall. Attitude determination. *Lecture notes at Virginia Tech, http://www.aoe.vt.edu/∼cdhall/courses/aoe4140/attde.pdf*.

[33] Y. Li and J. Yuan. Attitude determination using gps vector observations. Technical report, University of New South Wales, Sydney, Australia, and Northwestern Polytechnical University, Xi'an, China.

[34] http://www-stat.stanford.edu/ wavelab/.

[35] Wen-Hong Zhu and Tom Lamarche. Velocity estimation by using position and acceleration sensors. *Industrial Electronics, IEEE Transactions on*, 54:2706–2715, 2007.

[36] Felipe Bohorquez, Paul Samuel, Jayant Sirohi, Darryll Pines, Lael Rudd, and Ron Perel. Design, analysis and hover performance of a rotary wing micro air vehicle. *Journal of the American Helicopter Society*, 48(2):80–90, 2003.

[37] M. Tischler and M. Cauffman. Frequency-response method for rotorcraft system identification: Flight application to BO-105 coupled rotor/fuselage dynamics. *Journal of the American Helicopter Society*, 37(3):3–17, 1999.

[38] B. Mettler, M. Tischler, and T. Kanade. System identification of small-size unmanned helicopter dynamics. In *Proceedings of the AHS 55th Forum*, Montreal, Quebec, Canada, May 1999.

[39] Bernard Mettler. *Identification Modeling and Characteristics of Miniature Rotorcraft*. Kluwer Academics Publisher, Nowell, MA, 2003.

[40] Tischler M. B. and M.G. Cauffman. Frequency-response method for rotorcraft system identification: Flight applications to BO-105 coupled rotor/fuselage dynamics. *Journal of the American Helicopter Society*, 37(3):3–17, 1992.

[41] Tischler M. B. Advances in aircraft flight control. *Taylor and Francis, London*, 1996.

[42] S.K. Kim and D.M. Tilbury. Mathematical modeling and experimental identification of an unmanned helicopter robot with flybar dynamics. *Journal of Robotic Systems*, 21(3):95–116, 2004.

[43] M. Tischler and R. Remple. *Aircraft and Rotorcraft System Identification: Engineering Methods with Flight Test Examples*. American Institute for Aeronautics and Astronautics, 2006.

[44] M. Srinivasan, J.S. Chahl, K. Weber, S. Venkatesh M.G. Nagle, and S.W. Zhang. Robot navigation inspired by principles of insect vision. *Robotics and Autonomous Systems*, 26:203–216, 1999.

[45] M.V. Srinivasan and S.W. Zhang. Visual motor computations in insects. *Annual Review if Neuroscience*, 27:679–696, 2004.

[46] M.O. Franz and H.A. Mallot. Biomimetic robot navigation. *Robotics and Autonomous Systems*, 30:133–153, 2000.

[47] J. Santos-Victor and G. Sandini. Embedded visual behaviors for navigation. *Robotics and Autonomous Systems*, 19:299–313, 1997.

[48] D. Coombs, M. Herman, T.H. Hong, and M. Nashman. Real-time obstacle avoidance using central flow divergence, and peripheral flow. *IEEE Transactions on Robotics and Automation*, 14:49–59, 1998.

[49] T. Netter and N. Franceschini. A robotic aircraft that follows terrain using a neuromorphic eye. In *Proceedings of the IEEE/RSJ IROS Conference on Robots and Systems*. Lausanne, Switzerland, October 2002.

[50] L. Muratet, S. Doncieux, Y. Briere, and J. Meyer. A contribution to vision-based autonomous helicopter flight in urban environments. *Robotics and Autonomous Systems*, 50(4):195–209, 2005.

[51] J. Serres, D. Dray, F. Ruffier, and N. Franceschini. A vision-based autopilot for a miniature air vehicle: joint speed control and lateral obstacle avoidance. *Autonomous Robots*, 25:103–122, 2008.

[52] Stefan Hrabar and Gaurav S. Sukhatme. A comparison of two camera configurations for optic-flow based navigation of a uav through urban canyons. In *In EEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2673–2680, 2004.

[53] J. Santos-Victor, G. Sandini, F. Curroto, and S. Garibaldi. Divergent stereo in autonomous navigation - from bees to robots. *International Journal of Computer Vision*, 14:159–177, 1995.

[54] K. Weber, S. Venkatesh, and M.V. Srinivasan. Robot navigation inspired by principles of insect vision. *Robotics and Autonomous Systems*, 26:203–216, 1999.

[55] A. Argyros, D. Tsakiris, and C. Groyer. Biomimetic centering behavior for mobile robots with panoramic sensors. *IEEE Robotics and Automation Magazine*, pages 21–30, December 2004.

[56] J. Serres, F. Ruffier, and N. Franceschini. Two optic flow regulators for speed control and obstacle avoidance. In *Proceedings of the IEEE International Conference on Medical Robotics and Biomechatronics*, Pisa, Italy, February 2005.

[57] Bruno Herisse, Francois Xavier Russotto, Tarek Hamel, and Robert Mahony. Hovering flight and vertical landing control of a vtol unmanned aerial vehicle using optical flow. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1404–1409. Acropolis Convention Center, Nice, France, 2008.

[58] S. Griffiths, J. Saunders, A. Curtis, B. Barber, T. McLain, and R. Beard. Maximizing miniature aerial vehicles: Obstacle and terrain avoidance for mavs. *IEEE Robotics and Automation Magazine*, (September):34–42, 2006.

[59] Lance F. Tammero and Michael H. Dickinson. The influence of visual landscape on the free flight behavior of the fruit fly *Drosophila melanogaster*. *J. Exp. Biol.*, 205:327–343, 2002.

[60] Oh P. Y. Green, W.E. and G. Barrows. Flying insect inspired vision for autonomous aerial robot maneuvers in near-earth environments. In *Proceedings of the IEEE International Conference on Robotics and Automation*, New Orleans,LA, April 2004.

[61] R. Harrison. A biologically inspired analog ic for visual collision detection. *IEEE Transactions on Circuits and Systems*, 52(11):2308–2318, November 2005.

[62] J. C. Zufferey and D Floreano. Fly-inspired visual steering of an ultralight indoor aircraft. *IEEE Transactions on Robotics*, 22:137–146, 2006.

[63] M. Egelhaaf, R. Kern, H. Krapp, J. Kretzberg, R. Kurtz, and A. Warzecha. Neural encoding of behaviourally relevant visual-motion information in the fly. *Trends in Neurosciences*, 25:96–102, 2002.

[64] A. Borst and J. Haag. Neural networks in the cockpit of the fly. *J. Comp. Physiol. A*, 188:419–437, 2002.

[65] H.G. Krapp, B. Hengstenberg, and R. Hengstenberg. Dendritic structure and receptive-field organization of optic flow processing interneurons in the fly. *J. Neurophysiol.*, 79:1902–1917, 1998.

[66] M.A. Frye and M.H. Dickinson. Fly flight: A model for the neural control of complex behavior. *Neuron*, 32:385–388, 2001.

[67] J. S. Humbert, R. M. Murray, and M. H. Dickinson. A control-oriented analysis of bio-inspired visuomotor convergence. In *Proceedings of the 44th IEEE Conference on Decision and Control*. Seville, Spain, 2005.

[68] J. S. Humbert, A. M. Hyslop, and M. W. Chinn. Experimental validation of wide-field integration methods for autonomous navigation. In *Proceedings of the IEEE Conference on INtelligent Robots and Systems (IROS)*. San Diego, CA, 2007.

[69] J S Humbert and A H Hyslop. Bio-inspired visuomotor convergence. *IEEE Transactions on Robotics (in press)*, 2009.

[70] J. Conroy and D. Pines. System identification of a miniature electric helicopter using mems inertial, optic flow, and sonar sensing. In *Proceedings of the American Helicopter Society*. Virginia Beach, VA, 2007.

[71] G. Gremillion and J.S. Humbert. System identification of a quadrotor micro air vehicle. *AIAA Conference on Atmospheric Flight Mechanics, 2010*, 2010.

[72] Karin Nordstrm and David C. O'Carroll. Feature detection and the hypercomplex property in insects. *Trends in Neurosciences*, 32(7):383 – 391, 2009.

[73] T.S. Collett and J. Zeil. *The Selection and Use of Landmarks by Insects.*, pages 41–65. 1997.

[74] Wolfgang Strzl and Jochen Zeil. Depth, contrast and view-based homing in outdoor scenes. *Biological Cybernetics*, 96:519–531, 2007.

[75] R. Voss and J. Zeil. Active vision in insects: an analysis of object-directed zig-zag flights in wasps (odynerus spinipes?, eumenidae). *Journal of Comparative Physiology A: Neuroethology, Sensory, Neural, and Behavioral Physiology*, 182:377–387, 1998.

[76] Longuet-Higgins H. A computer algorithm for reconstructing a scene from two projections. *Nature*, 293:133–135.

[77] K. Prazdny. On the information in optical flows. *Computer Vision, Graphics, and Image Processing*, 22(2):239 – 259, 1983.

[78] R. Tsai and T. Huang. Estimating three-dimensional motion parameters of a rigid planar patch. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 29(6):1147 – 1152, dec 1981.

[79] Ted J. Broida and Rama Chellappa. Estimation of object motion parameters from noisy images. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, PAMI-8(1):90 –99, jan. 1986.

[80] T.J. Broida and R. Chellappa. Estimating the kinematics and structure of a rigid object from a sequence of monocular images. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 13(6):497 –513, jun. 1991.

[81] Ting-Hu Wu, Rama Chellappa, and Qinfen Zheng. Experiments on estimating egomotion and structure parameters using long monocular image sequences. *International Journal of Computer Vision*, 15:77–103, 1995.

[82] Gang Qian and Rama Chellappa. Structure from motion using sequential monte carlo methods. *International Journal of Computer Vision*, 59:5–31, 2004.

[83] Gang Qian, R. Chellappa, and Qinfen Zheng. Bayesian algorithms for simultaneous structure from motion estimation of multiple independently moving objects. *Image Processing, IEEE Transactions on*, 14(1):94 –109, jan. 2005.

[84] A.K. Roy-Chowdhury and R. Chellappa. Statistical bias in 3-d reconstruction from a monocular video. *Image Processing, IEEE Transactions on*, 14(8):1057 –1062, aug. 2005.

[85] S. Soatto and P. Perona. Reducing "structure from motion": a general framework for dynamic vision. part 1: modeling. *International Journal of Computer Vision*, 20(9):993–942, September 1998.

[86] S. Soatto and P. Perona. Reducing "structure from motion": a general framework for dynamic vision. part 2: Implementation and experimental assessment. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(9):943–960, September 1998.

[87] Jian Li and R. Chellappa. Structure from planar motion. *Image Processing, IEEE Transactions on*, 15(11):3466 –3477, nov. 2006.

[88] Kalle Åström and Magnus Oskarsson. Solutions and ambiguities of the structure and motion problem for 1dretinal vision. *J. Math. Imaging Vis.*, 12(2):121–135, 2000.

[89] Marco Zucchelli. Optical flow based structure from motion. Technical report, in Numerical Analysis and Computer Science. Stockholm: Kungl Tekniska Hogskolan (Royal Institute of Techology, 2002.

[90] Juyang Weng, Narendra Ahuja, and Thomas S. Huang. *Motion and Structure from Image Sequences*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 1992.

[91] A. Azarbayejani, B. Horowitz, and A. Pentland. Recursive estimation of structure and motion using relative orientation constraints. pages 294 –299, jun. 1993.

[92] A. Azarbayejani and A.P. Pentland. Recursive estimation of motion, structure, and focal length. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 17(6):562 –575, jun. 1995.

[93] Ashwin Dani and Warren Dixon. Single camera structure and motion estimation. In Graziano Chesi and Koichi Hashimoto, editors, *Visual Servoing via Advanced Numerical Methods*, volume 401 of *Lecture Notes in Control and Information Sciences*, pages 209–229. Springer Berlin / Heidelberg, 2010.

[94] S. Gupta, D. Aiken, G. Hu, and W.E. Dixon. Lyapunov-based range and motion identification for a nonaffine perspective dynamic system. page 6 pp., jun. 2006.

[95] Cornelia Fermller and Yiannis Aloimonos. Geometry of eye design: Biology and technology. Technical report, Proc. of the tenth International Workshop on Theoretical Foundations of Computer Vision: MultiImage Analysis (also appeared in Lecture Notes in Computer Science, 1998.

[96] R.F. Vassallo, J. Santos-Victor, and H.J. Schneebeli. A general approach for egomotion estimation with omnidirectional images. pages 97 – 103, 2002.

[97] Davide Scaramuzza and Agostino Martinelli. A flexible technique for accurate omnidirectional camera calibration and structure from motion. In *Proceedings of IEEE International Conference on Computer Vision Systems*, 2006.

[98] Steven Rey Ortiz, Ricardo Gutierrez-osuna, Nancy Amato, Reza Langari, and Valerie Taylor. Vision and certainty grids. (august 2004), 2004.

[99] Peng Chang and Martial Hebert. Omni-directional structure from motion. In *Proceedings of the 2000 IEEE Workshop on Omnidirectional Vision*, pages 127 – 133, July 2000.

[100] O. Dahl and A. Heyden. Provably stable nonlinear and adaptive observers for dynamic structure and motion estimation. pages 1 –4, Oct. 2007.

[101] O. Dahl and A. Heyden. Dynamic structure from motion based on nonlinear adaptive observers. pages 1–4, dec. 2008.

[102] Ashwin P. Dani, Nicholas R. Fischer, Zhen Kan, and Warren E. Dixon. Nonlinear observer for structure estimation using a paracatadioptric camera. pages 3487 –3492, jun. 2010.